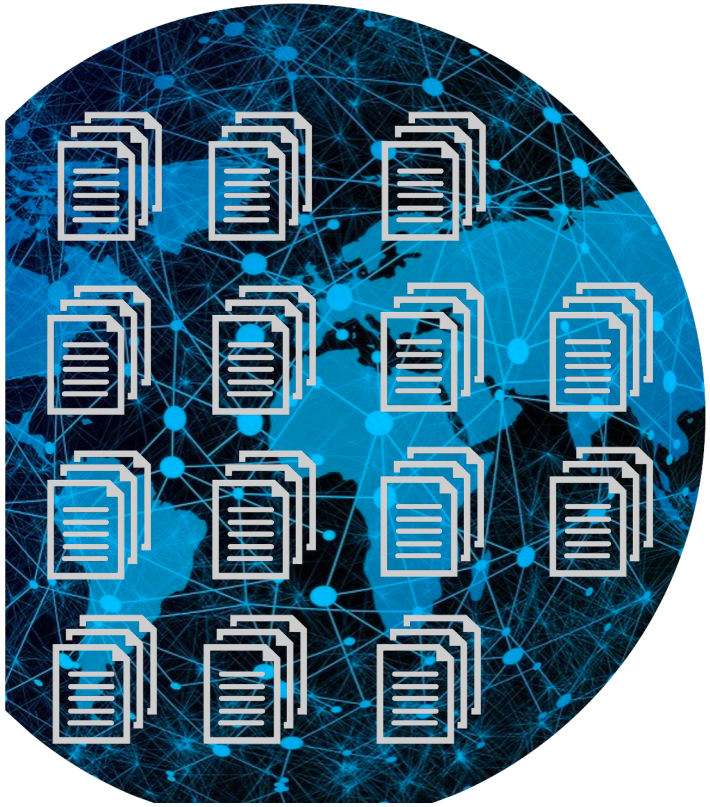
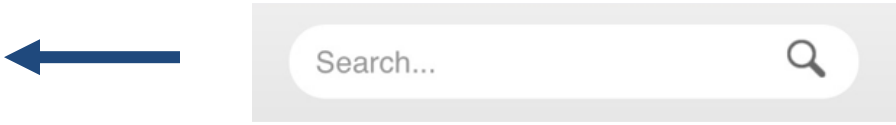
Neural Matching and Importance Learning in Information Retrieval

Zhuyun Dai
Language Technologies Institute
Carnegie Mellon University
zhuyund@cs.cmu.edu

Information Retrieval

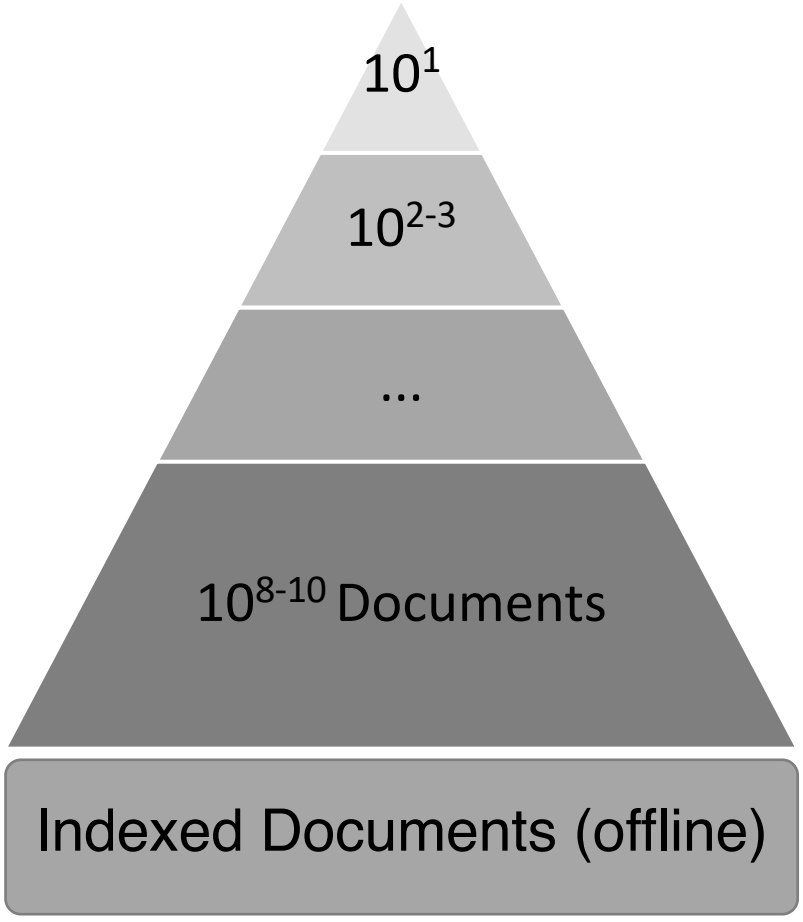


Find from MASSIVE documents…



A small set of relevant results

Today's Text Retrieval Systems



Reranking

Machine Learning Models
(LeToR, 2008)

Initial Retrieval

Heuristics
(BM25, 1994)

Indexing

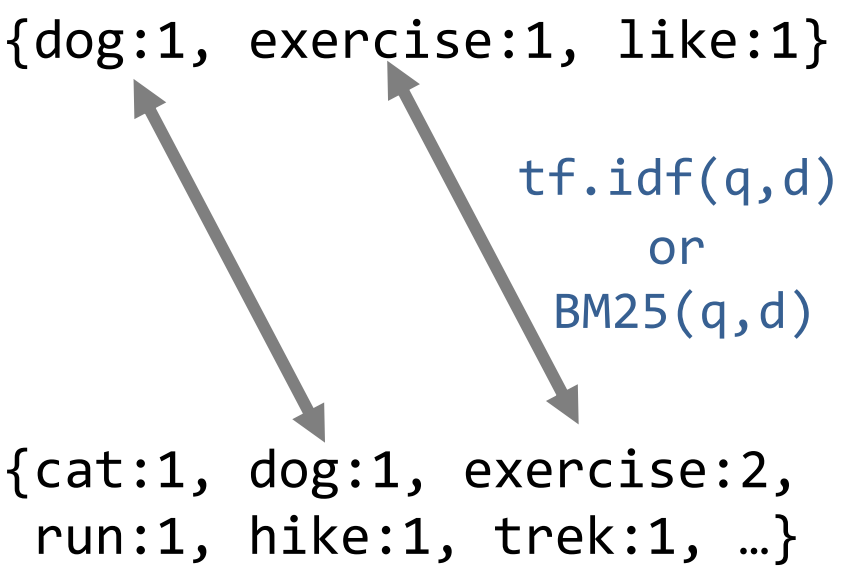
Store Processed Documents

Initial Ranking: Bag-of-Words Retrieval

Do dogs like exercising?

A Document

"Unlike cats, dogs are usually great exercise pals. Many breeds enjoy running and hiking, and will happily trek along on any trip. Exercise time may vary…"



Reranking: Learning-To-Rank (LeToR)

Do dogs like exercising?

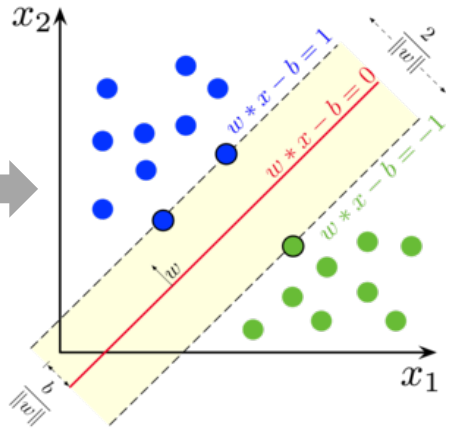
A Document

"Unlike cats, dogs are usually great exercise pals. Many breeds enjoy running and hiking, and will happily trek along on any trip. Exercise time may vary…"

Feature Vector

```
BM25(q,d)
tfidf(q,d)
len(d)
spam(d)
len(q)
…
```

Machine Learning



Bottlenecks in Bag-of-Words Retrieval

Query-Document Matching:

Exact Lexical Match

40-50% query-document pairs has a vocabulary mismatch[1]

Query/document Representation:

Frequency Weighting

50-260% improvements can be made with oracle term weights[1]

[1] Le Zhao. Modeling and solving term mismatch for full-text retrieval. PhD thesis, Carnegie Mellon University, 2012.

Does IR need deeper language understanding?

Decades of efforts to use more complex NLP in large-scale IR

- POS, Parsing, Neural Networks (Late 1980s)
- Difficult to train: curse of dimensionality
- Slow & Do not really improve Bag-of-Words baselines

Deep Learning

- Better Models, More Data & Faster Computation
- This time may be different

This Dissertation

My dissertation research aims to leverage the advantages of neural networks to improve language understanding in today's information retrieval systems.

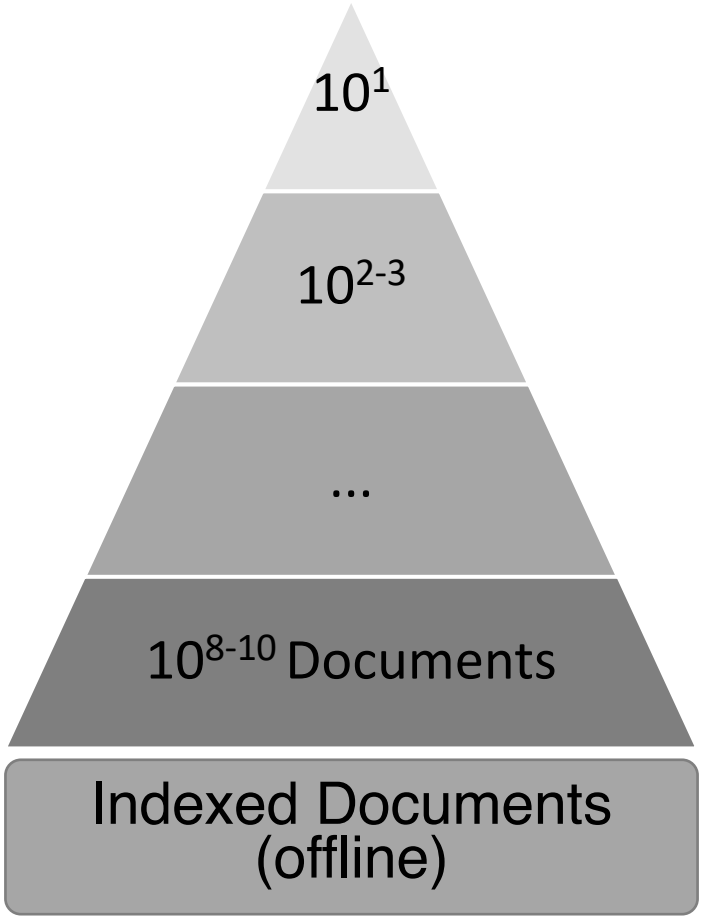
Query-Document Matching:

Exact Lexical Match

Query/document Representation:

Frequency Weighting

This Talk



PART I: MATCHING

Kernel-Based Ranking

Bring Soft Match into Ranking

BERT-Based Ranking

Bring General Language Understanding into Matching

PART II: REPRESENTATION

Context-Aware Term Weighting

Bring Deeper Language Understanding into Initial Retrieval

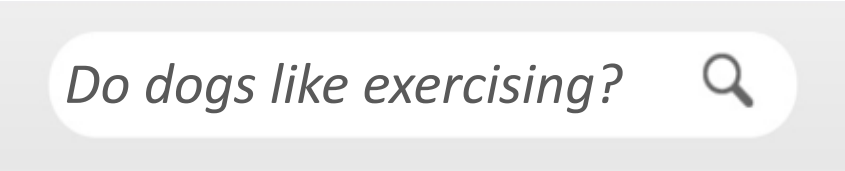
KERNEL-BASED RANKING

Bring Soft Match into Ranking

[X*__D__*CPL SIGIR'17]
[__D__XCL WSDM'18]
[PAC__D__XC SIGIR'18[s]]
[__D__FRC WWW'19]

*: Equal Contribution, [s]: Short Paper

Vocabulary Mismatch



{dog:1, exercise:1, like:1}

A Relevant Document

"Unlike cats, dogs are usually great exercise pals. Many breeds enjoy running and hiking, and will happily trek along on any trip. Exercise time may vary…"

{cat:1, dog:1, exercise:2, run:1, hike:1, trek:1, …}

Exact Lexical Match

Vocabulary Mismatch

puppies

Do ~~dogs~~ like exercising?

A Relevant Document

"Unlike cats, dogs are usually great exercise pals. Many breeds enjoy running and hiking, and will happily trek along on any trip. Exercise time may vary…"

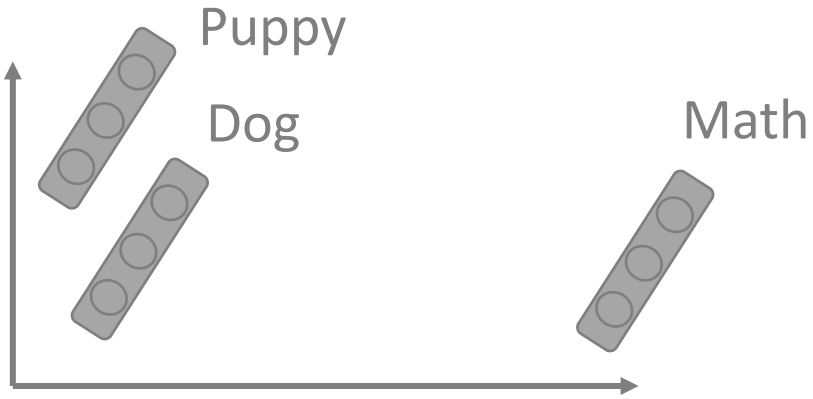
puppies

{~~dog~~:1, exercise:1, like:1}

{cat:1, dog:1, exercise:2,
run:1, hike:1, trek:1, …}

Exact Lexical Match

Bridge the Vocabulary Mismatch: Continuous Representations

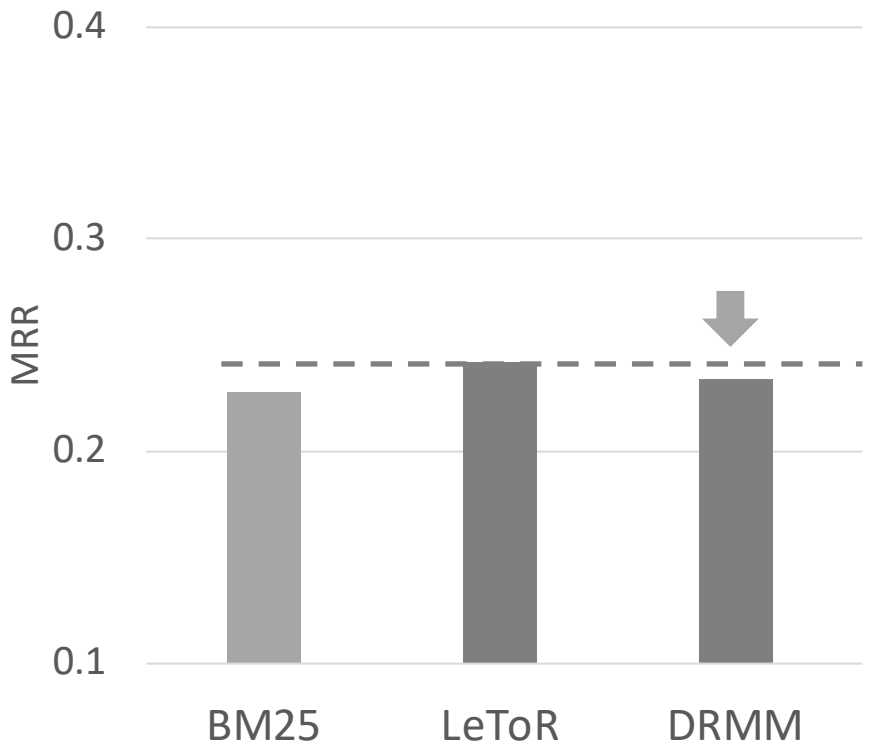


Not a new idea in IR

- LSI, LSA, LDA, MatchPlus, …
- Word2Vec, GloVE, …
- DSSM (Huang et al, 2013), CDSSM (Shen et al, 2014), ARC-II (Hu et al, 2014), DRMM (Guo et al, 2016)

Results for large-scale IR were mixed

Bridge the Vocabulary Mismatch: Continuous Representations

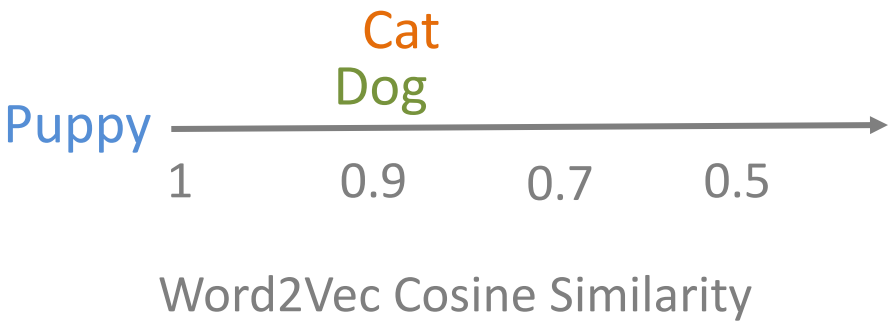


Dataset: Sogou-Log, re-rank top 10-30
Training: a search log of 100K queries
Testing: 1K queries (Testing-RAW set)

DRMM [Guo et al. 2016]

- SOTA neural ranker
- Uses Word2Vec
- Beats (unsupervised) BM25, but not supervised LeToR

Soft Match Signals are Noisy and Mixed

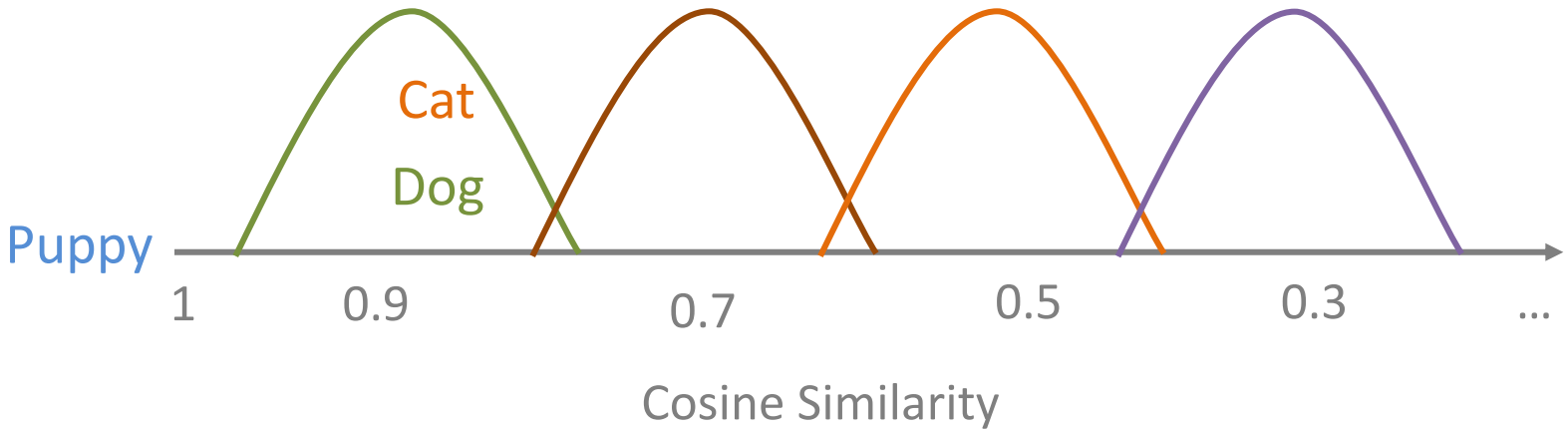


Do puppies like workout? 🔍

Doc 1: "Dogs like workout." 😄

Doc 2 :"Cats do not like workout." 😡

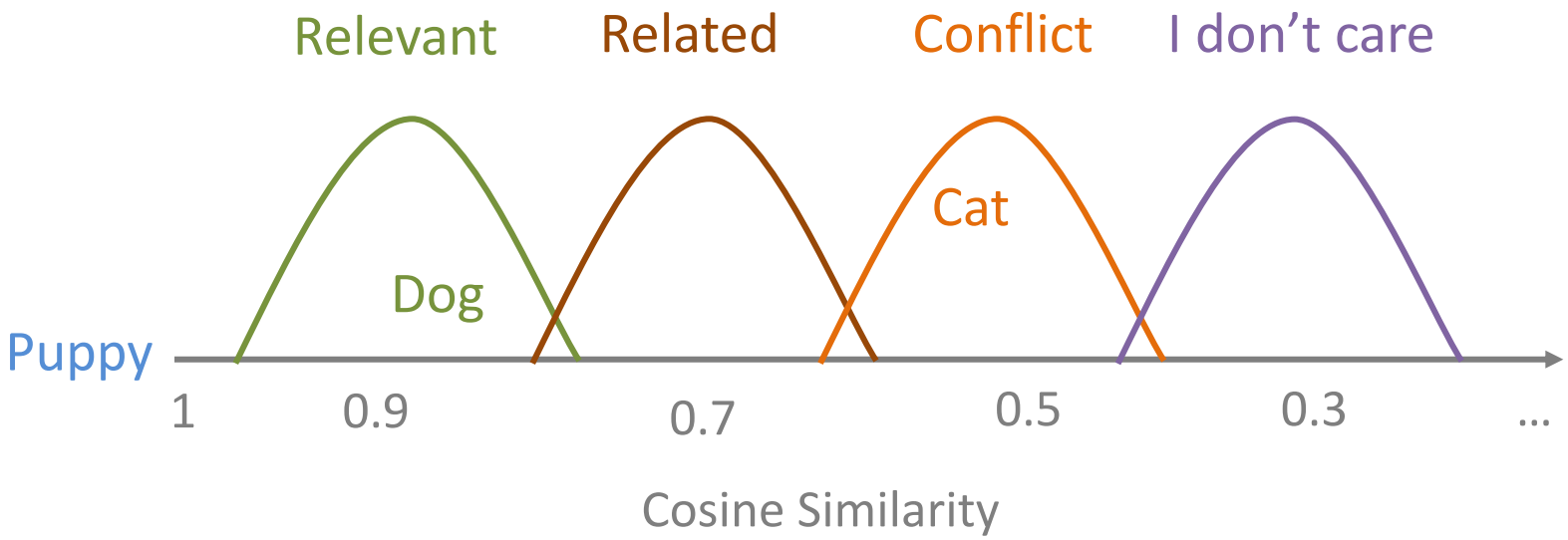
Kernel-Pooling: Learn to group soft matches by contribution to relevance



Soft-Match Kernels:

$(\mu = 0.9, \sigma = 0.2), (\mu = 0.7, \sigma = 0.2), (\mu = 0.5, \sigma = 0.2), \ldots$

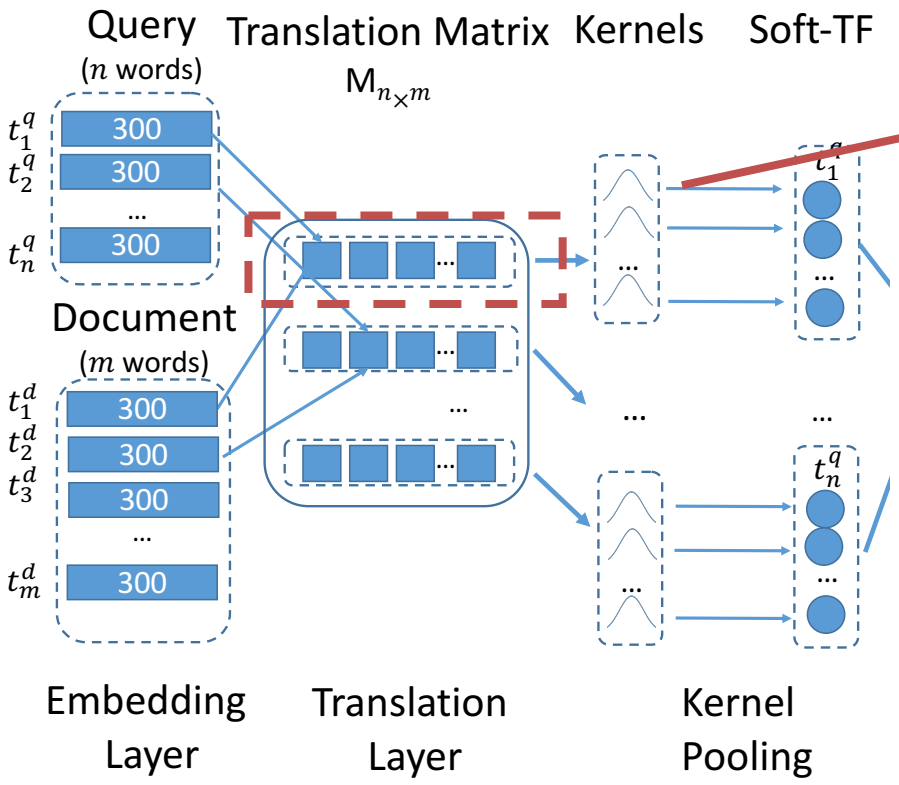
Kernel-Pooling: Learn to group soft matches by contribution to relevance

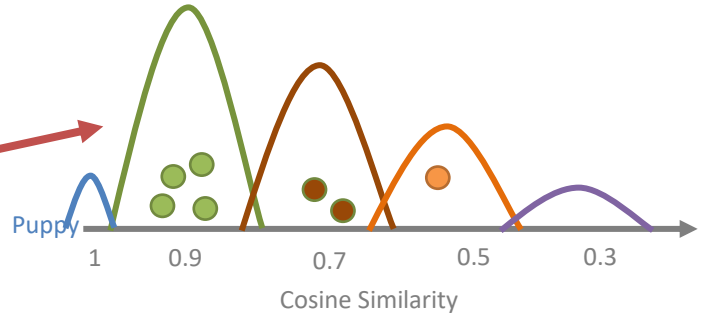


Soft-Match Kernels:

$(\mu = 0.9, \sigma = 0.2), (\mu = 0.7, \sigma = 0.2), (\mu = 0.5, \sigma = 0.2), \ldots$

K-NRM: Kernel-based Neural Ranking Model

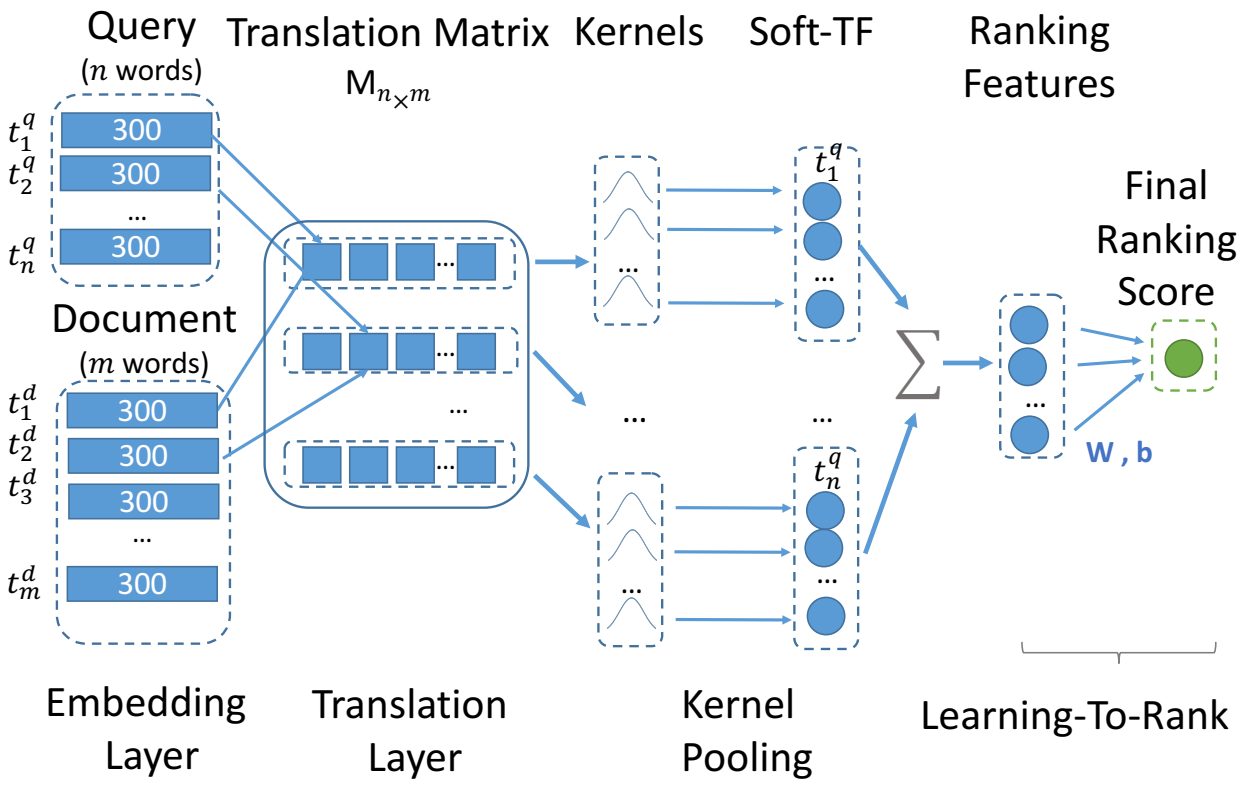




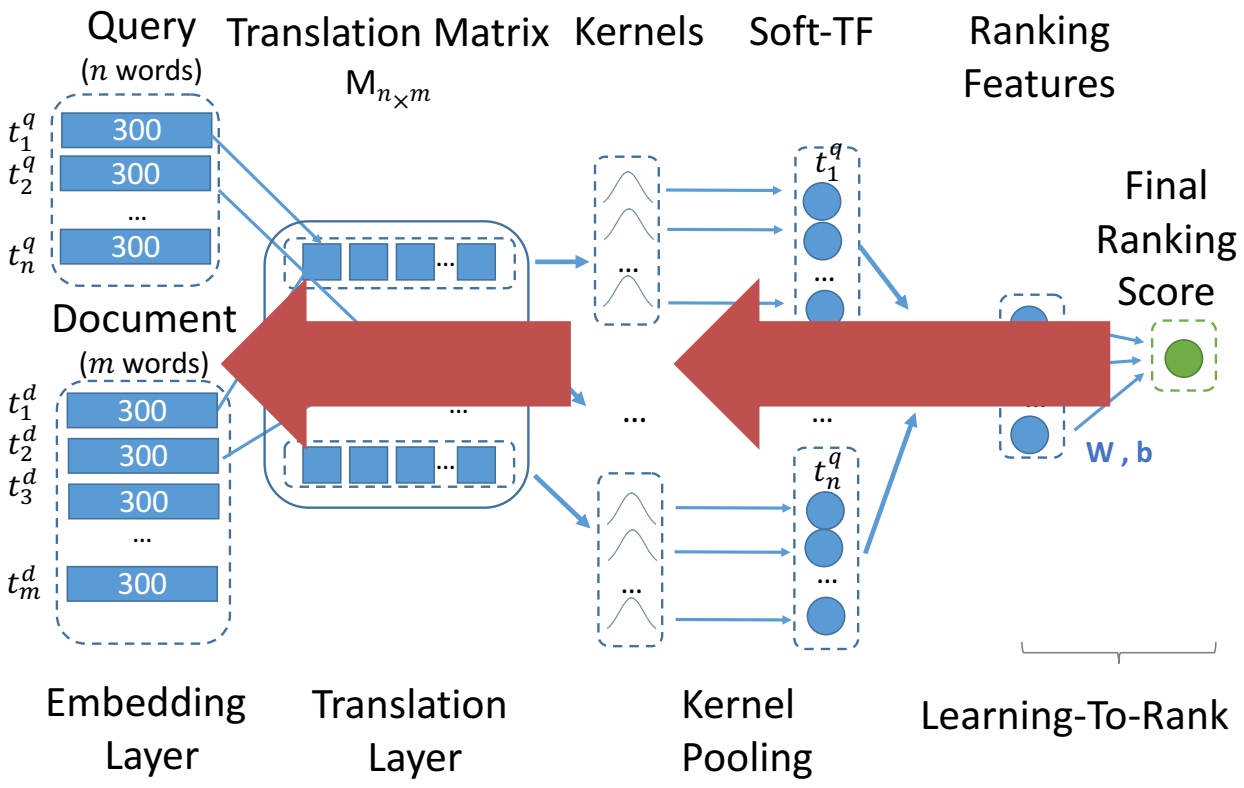
Soft Term Frequency (TF) in Kernels

$$K_k(M_i) = \sum_{j=1}^{m} \exp\left(-\frac{M_{ij} - \mu_k}{2{\sigma_k}^2}\right)$$

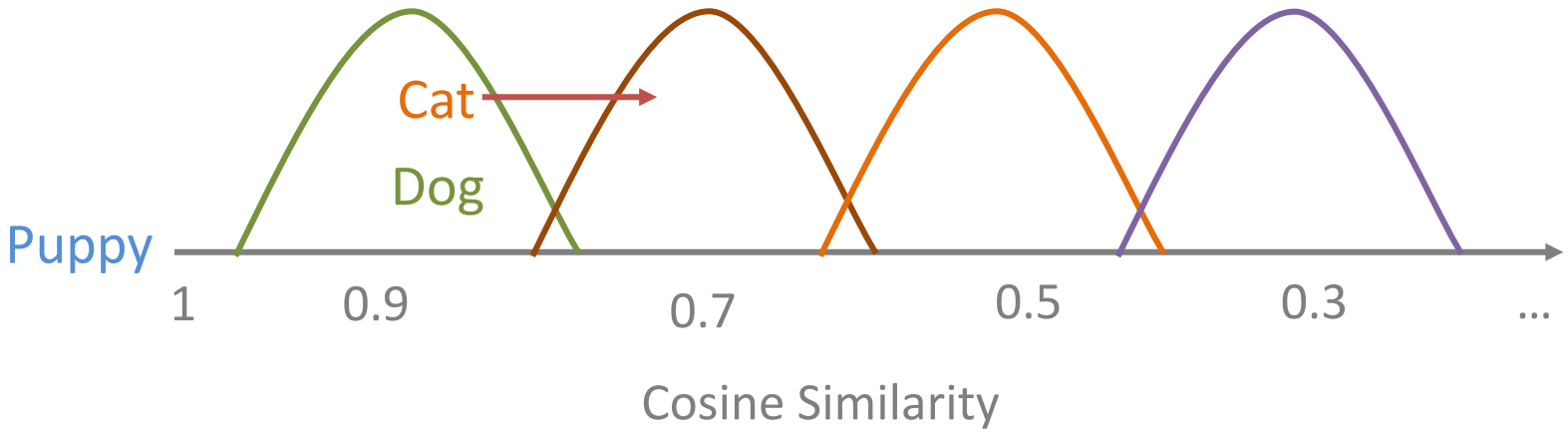
K-NRM: Kernel-based Neural Ranking Model



End-To-End Train K-NRM



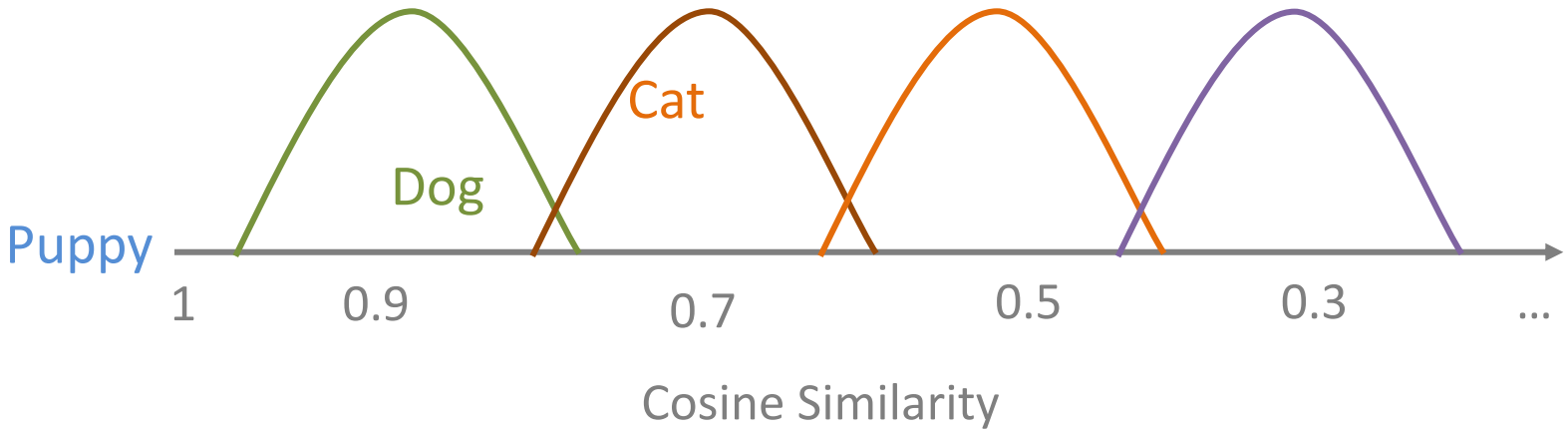
Kernel-Pooling: Learn to group soft matches by contribution to relevance



Soft-Match Kernels:

$(\mu = 0.9, \sigma = 0.2), (\mu = 0.7, \sigma = 0.2), (\mu = 0.5, \sigma = 0.2), \ldots$

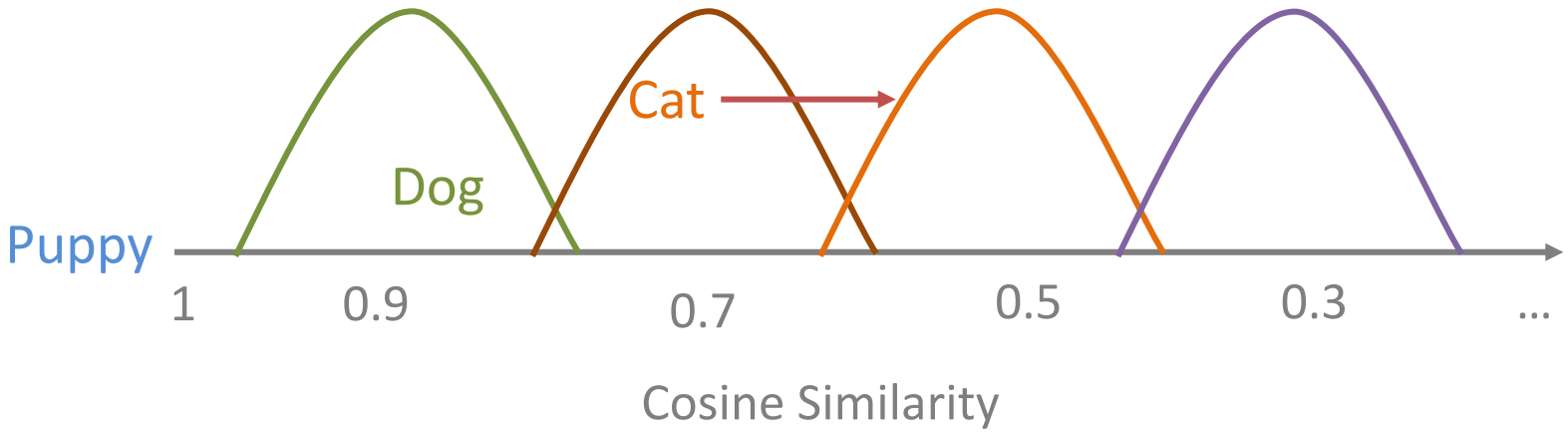
Kernel-Pooling: Learn to group soft matches by contribution to relevance



Soft-Match Kernels:

$(\mu = 0.9, \sigma = 0.2), (\mu = 0.7, \sigma = 0.2), (\mu = 0.5, \sigma = 0.2), \ldots$

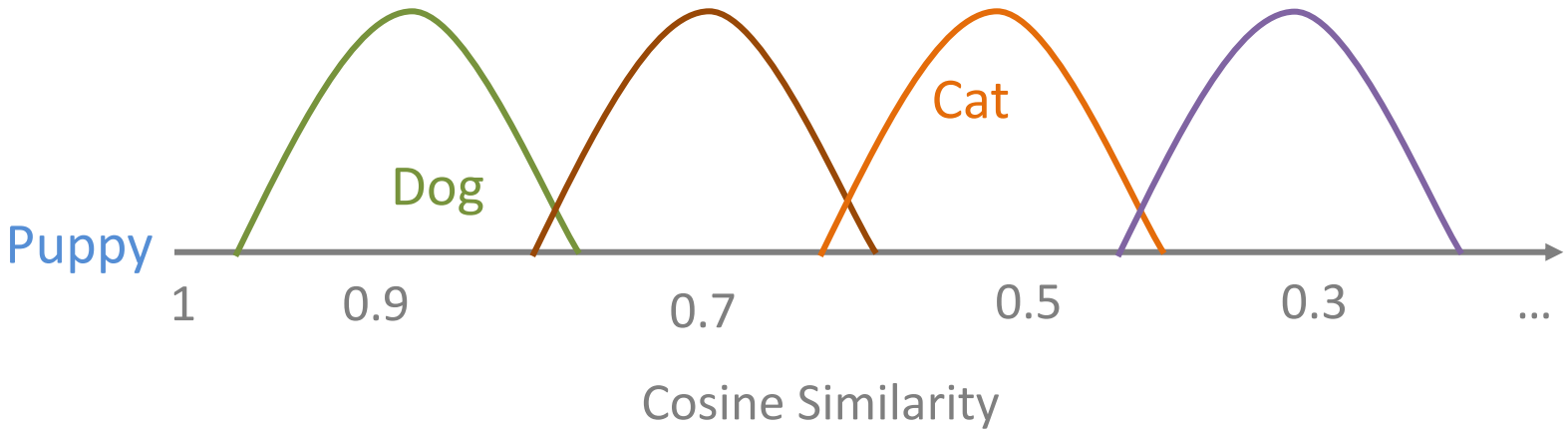
Kernel-Pooling: Learn to group soft matches by contribution to relevance



Soft-Match Kernels:

$(\mu = 0.9, \sigma = 0.2), (\mu = 0.7, \sigma = 0.2), (\mu = 0.5, \sigma = 0.2), \ldots$

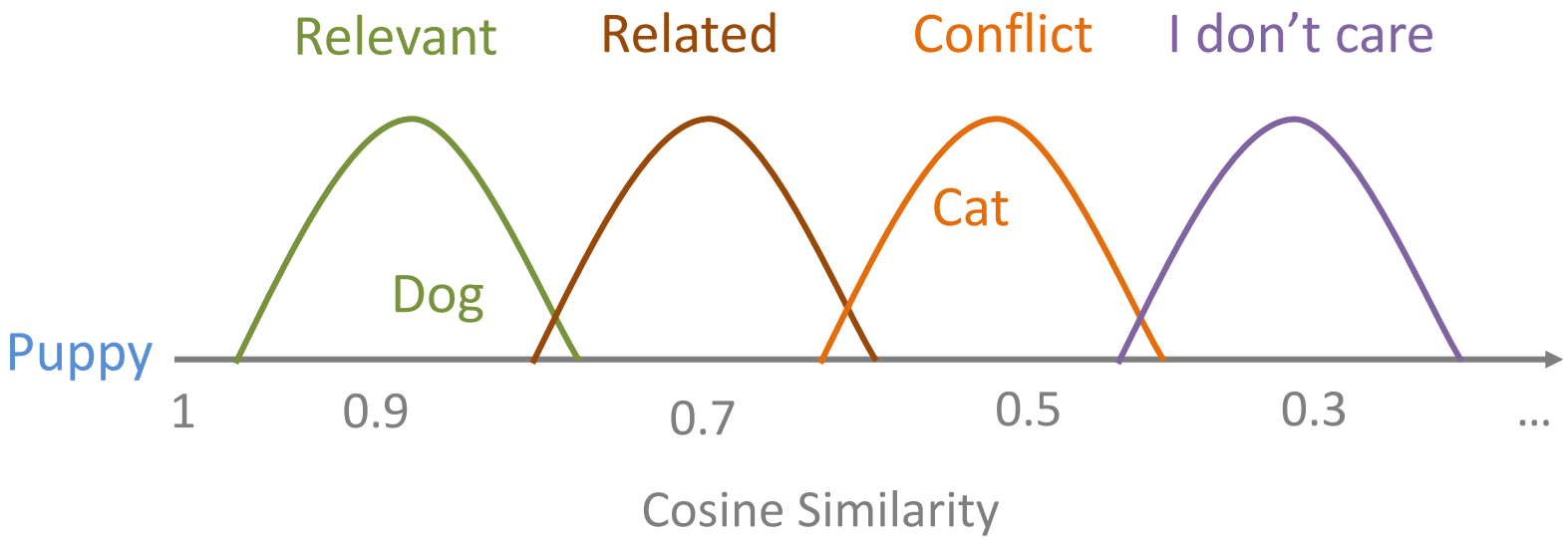
Kernel-Pooling: Learn to group soft matches by contribution to relevance



Soft-Match Kernels:

$(\mu = 0.9, \sigma = 0.2), (\mu = 0.7, \sigma = 0.2), (\mu = 0.5, \sigma = 0.2), \ldots$

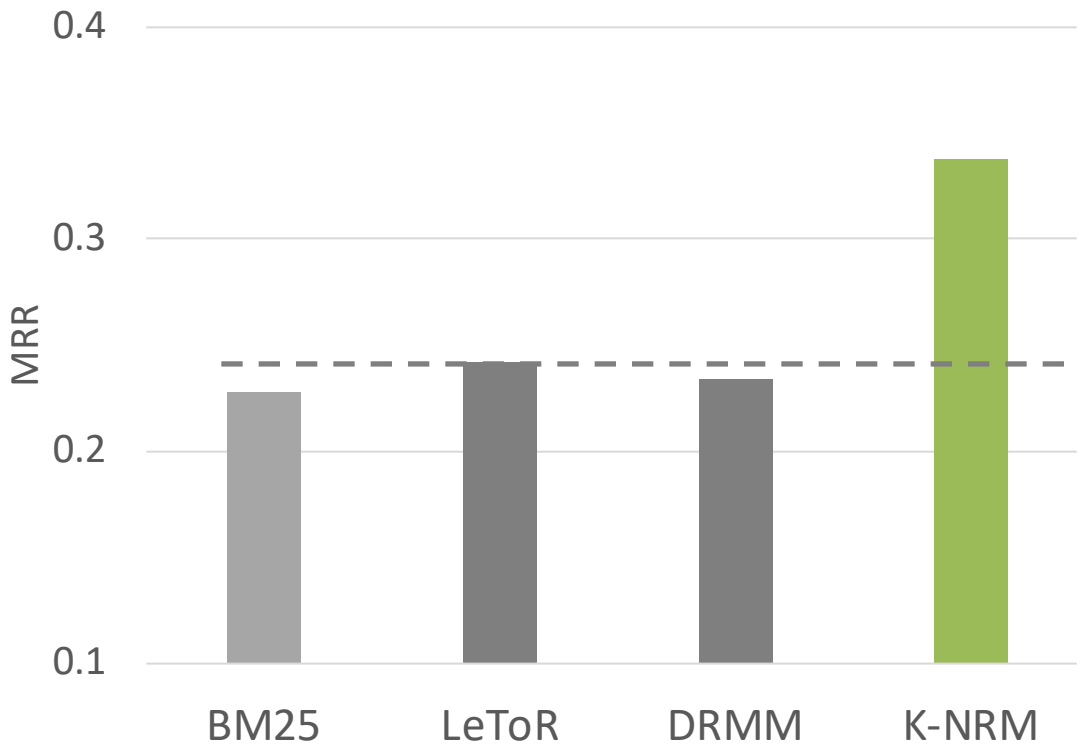
Kernel-Pooling: Learn to group soft matches by contribution to relevance



Soft-Match Kernels:

$(\mu = 0.9, \sigma = 0.2), (\mu = 0.7, \sigma = 0.2), (\mu = 0.5, \sigma = 0.2), \ldots$

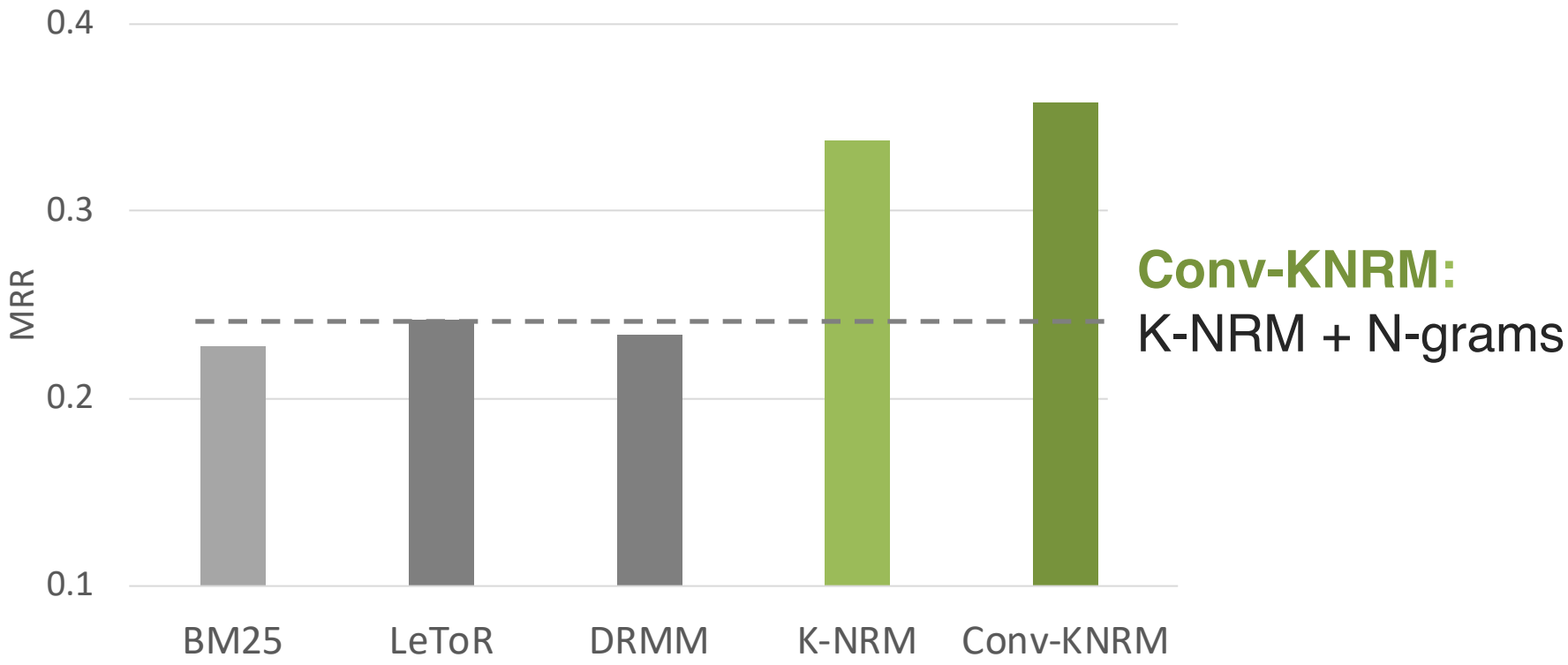
Effectiveness of Kernel-based Ranking



K-NRM:
Word-word similarity is aligned with relevance

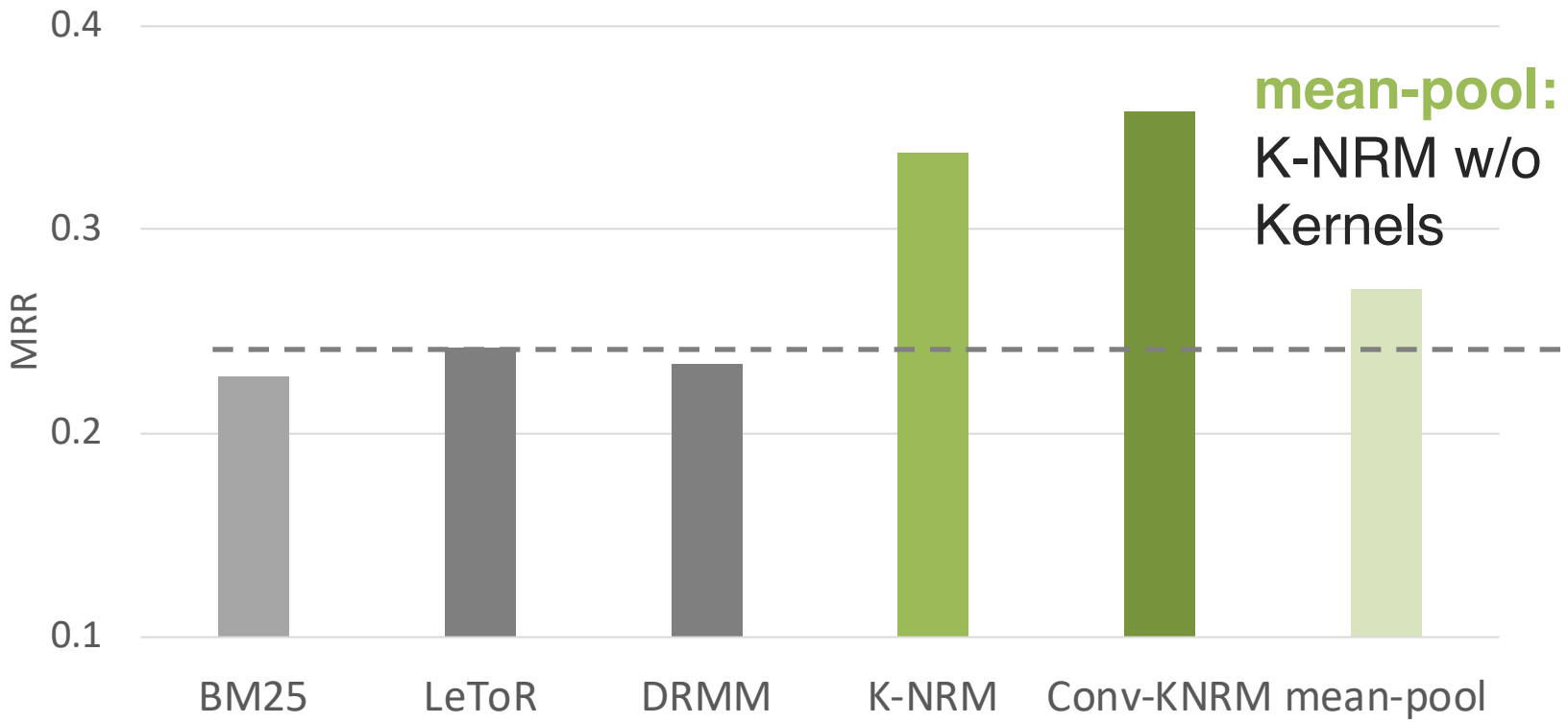
Dataset: Sogou-Log, re-rank top 10-30
Training: a search log of 100K queries
Testing: 1K queries (Testing-RAW set)

Effectiveness of Kernel-based Ranking



Dataset: Sogou-Log, re-rank top 10-30
Training: a search log of 100K queries
Testing: 1K queries (Testing-RAW set)

Effectiveness of Kernel-based Ranking

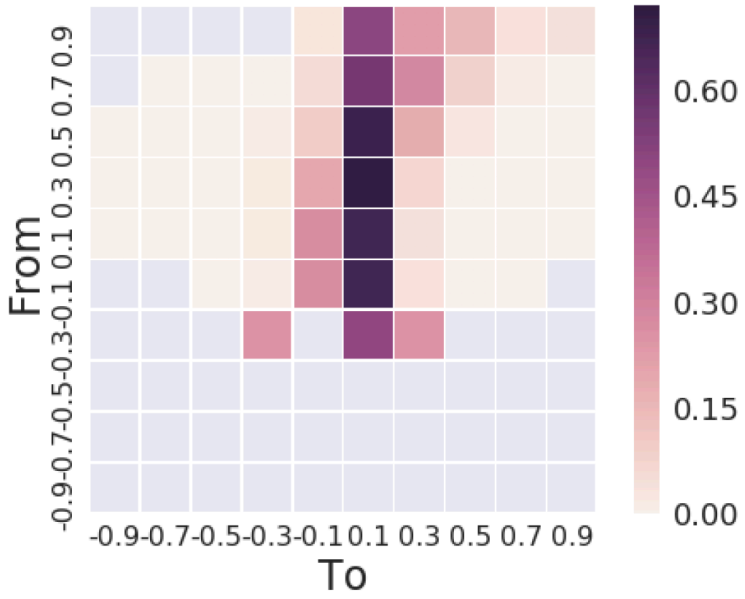


Dataset: Sogou-Log, re-rank top 10-30
Training: a search log of 100K queries
Testing: 1K queries (Testing-RAW set)

How Kernels Groups Soft Matches

Similarity != Relevance

- 58% word pairs **moved** kernels before/after training
- 90% word pairs considered similar by word2vec were **decoupled** by KNRM
  e.g., (wife, husband)
- New soft matches were **discovered**
  e.g., (pdf, reader)



Summary

Kernel-Based Ranking: Effective soft match for ranking

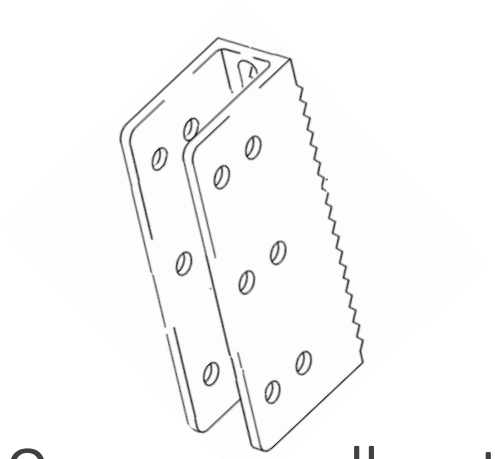
- One of the first neural models to outperform learning-to-rank w/ exact lexical matching features (+30%)
- Extend to n-grams, domain adaptation, non-text ranking

Change how people understand soft match in IR

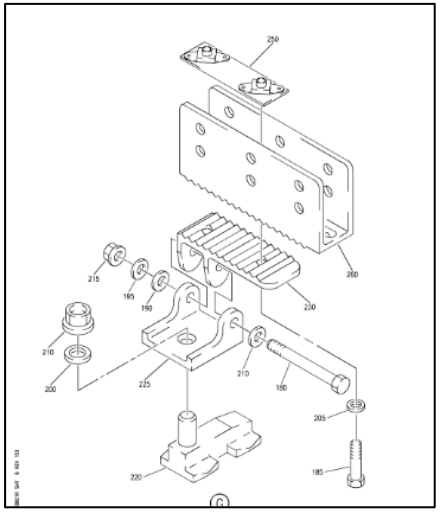
-Prior Research: words with similar linguistic uses in a corpus are also relevant in search

-We show: language usage in search can be very different

Generalizing to Non-Text Search Tasks



Query: a small part



Document: a large assembly

Engineering Diagram Retrieval: Boeing & Ikea

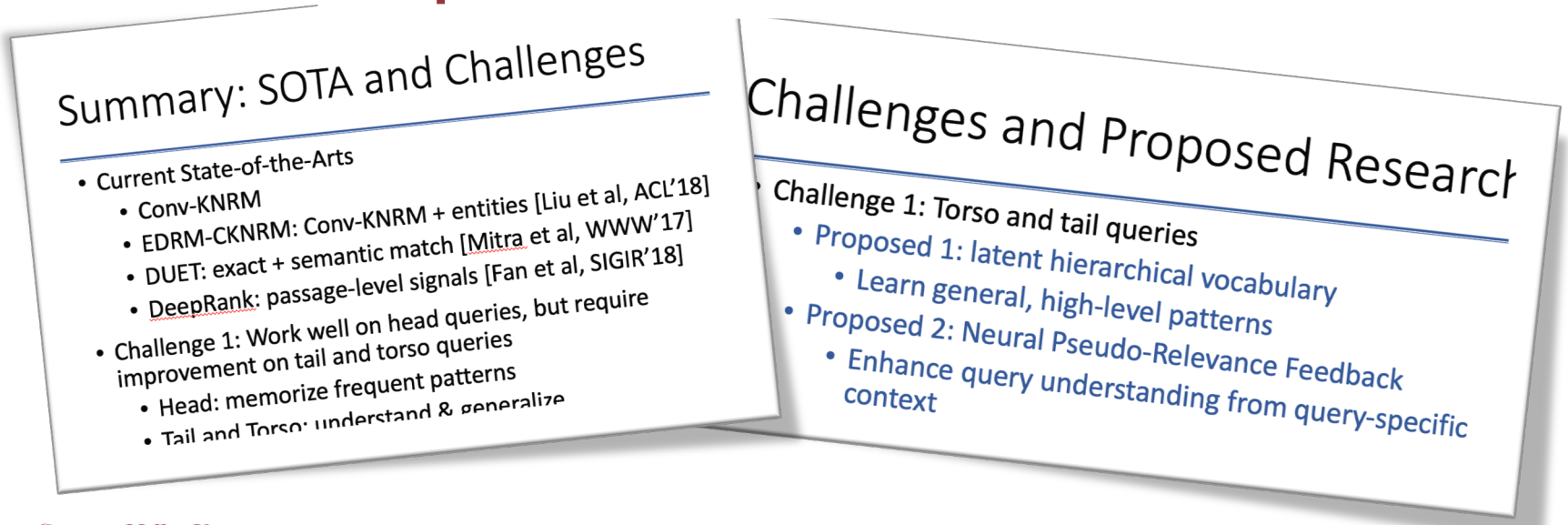
- Conv-KNRM architecture is also effective

Challenges

Work well when having rich training data, but need improvements under low-resource settings

- High-resource: memorize common search patterns
- Low-resource: understand & generalize

Proposed Solutions in 2018



Challenges

Work well when having rich training data, but need improvements under low-resource settings

- High-resource: memorize common search patterns
- Low-resource: understand & generalize

Pre-trained Deep Language Models!

BERT-BASED RANKING

Bring General Language Understanding into Ranking

[**D**C SIGIR'19[s]]

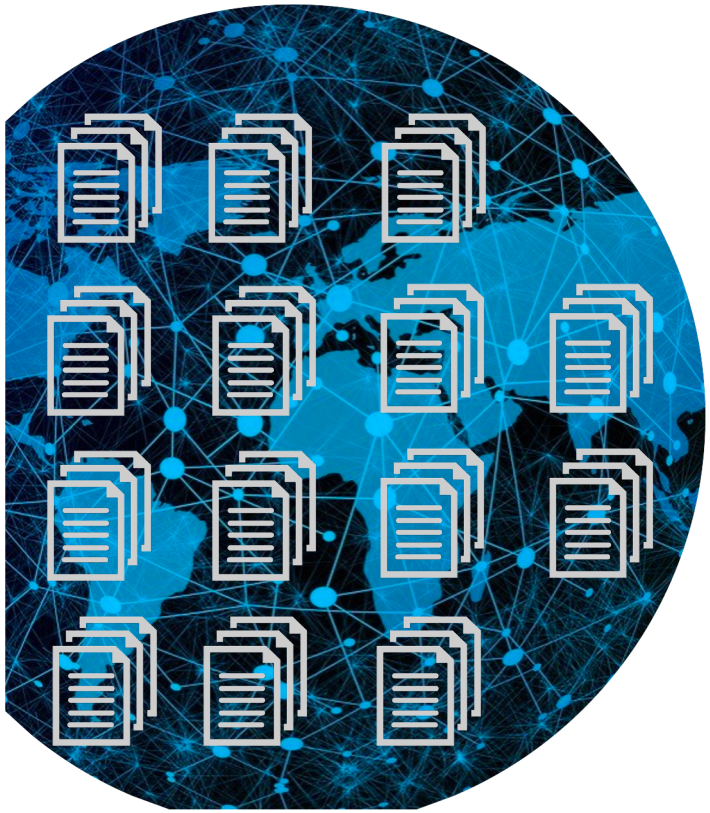
[s]: Short Paper

Yann LeCun's Cake Analogy

Specific Knowledge about the Search Task (icing)

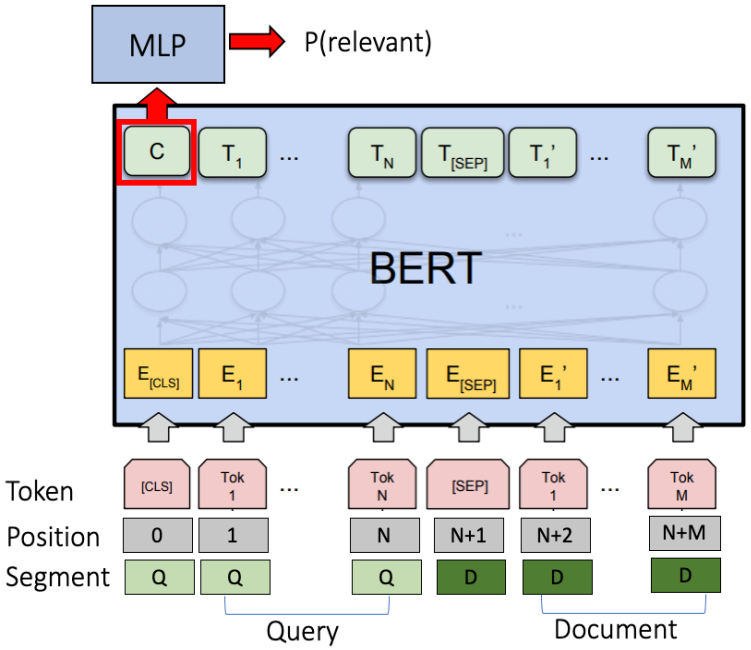
General Knowledge about the Language (cake)

General Language Understanding



Pre-train: Learn general language patterns from massive documents

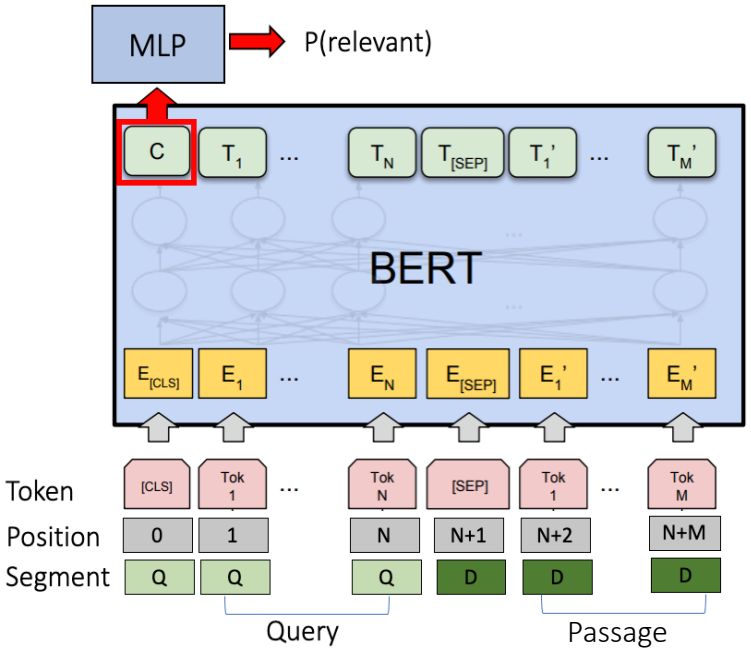
DocBERT: BERT for Document Reranking



Documents are too long for BERT

- Up to 512 tokens

DocBERT: BERT for Document Reranking



Documents are too long for BERT

- Up to 512 tokens

Split into Passages

- $concat(title, p_i)$: provides global context to passage p_i

Combine Passage Scores

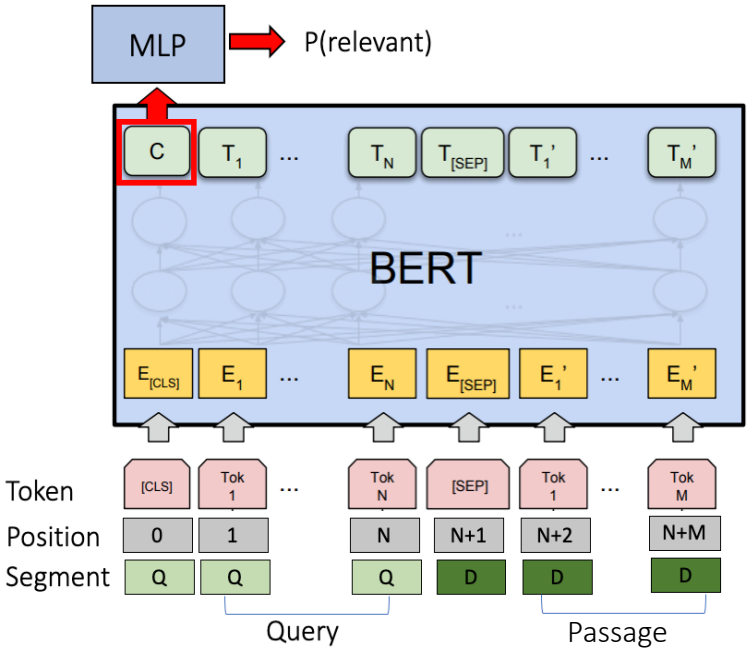
- DocBERT-FirstP:

$$\text{Rel}(\text{q}, \text{doc}) = BERT(q, p_0)$$

- DocBERT-maxP:

$$\text{Rel}(\text{q}, \text{doc}) = \max_i BERT(q, p_i)$$

DocBERT: BERT for Document Reranking



Documents are too long for BERT

- Up to 512 tokens

Split into Passages

- $concat(title, p_i)$: provides global context to passage p_i

Combine Passage Scores

- DocBERT-FirstP:

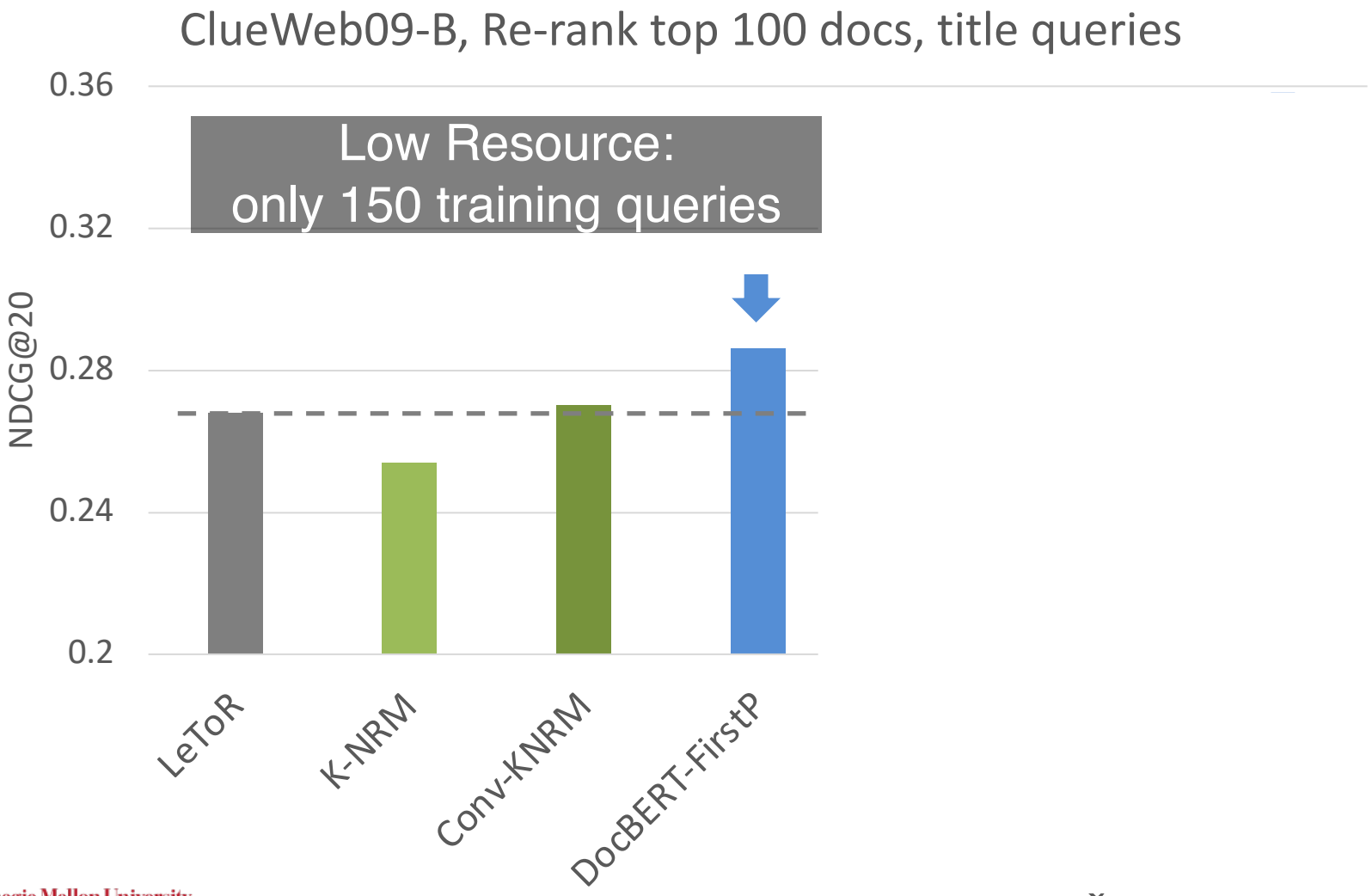
$$\text{Rel}(\text{q}, \text{doc}) = BERT(q, p_0)$$

- DocBERT-maxP:

$$\text{Rel}(\text{q}, \text{doc}) = \max_i BERT(q, p_i)$$

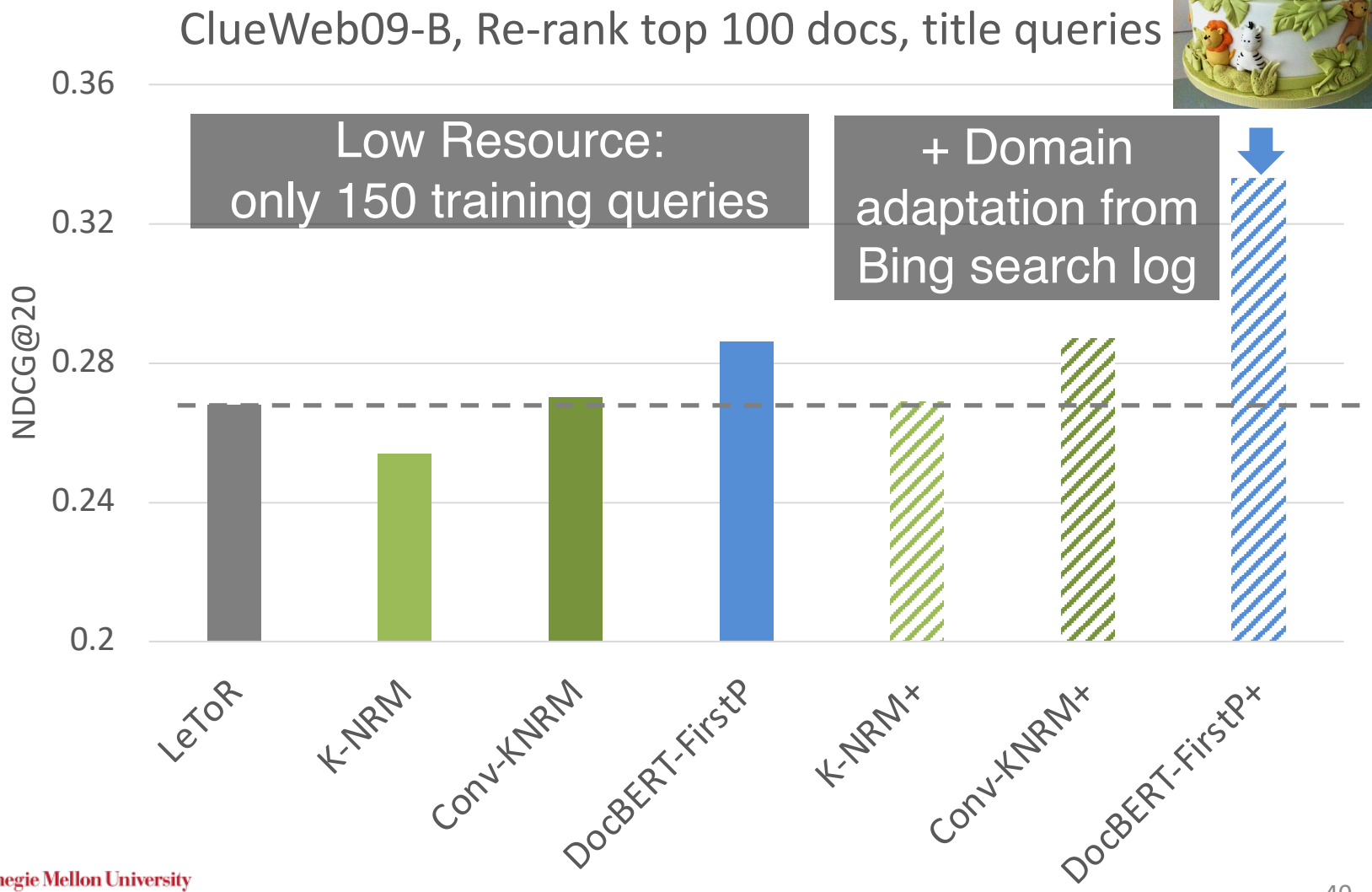
Train: $\text{label}(\text{q}, p_i) = \text{label}(q, doc)$

Effectiveness of “Cake”

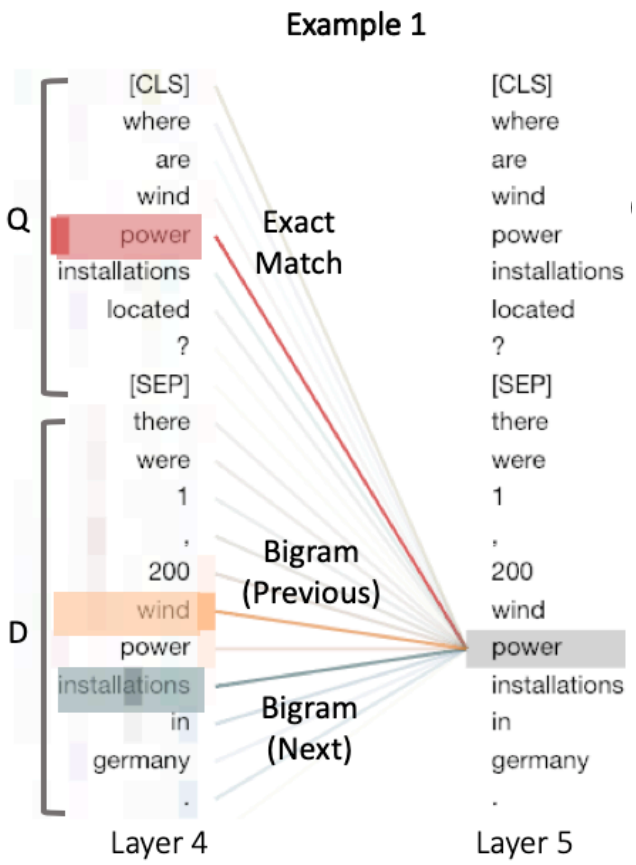


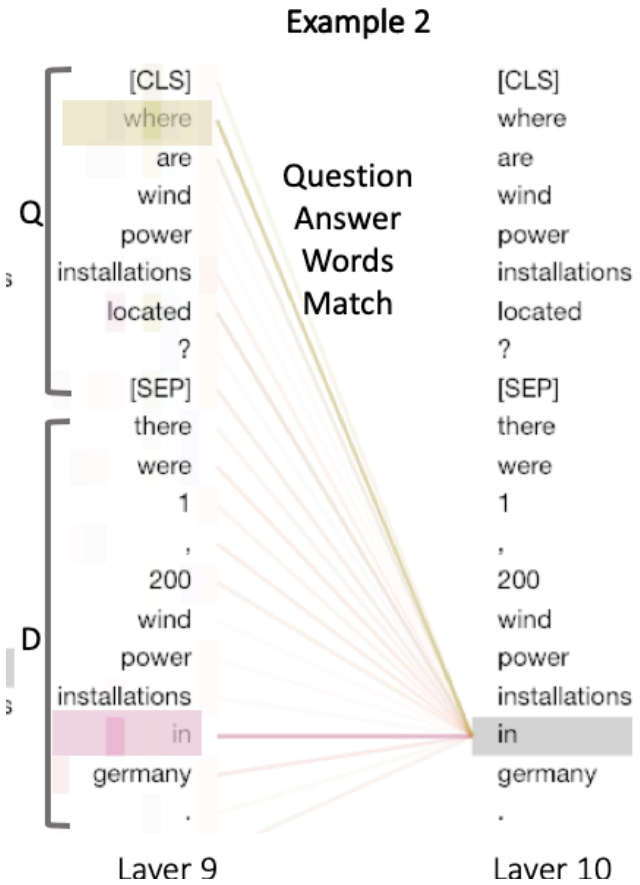
Add “Icing”


baby


ClueWeb09-B, Re-rank top 100 docs, title queries
Low Resource:
only 150 training queries
+ Domain
adaptation from
Bing search log
NDCG@20
0.36
0.32
0.28
0.24
0.2
LeToR
K-NRM
Conv-KNRM
DocBERT-FirstP
K-NRM+
Conv-KNRM+
DocBERT-FirstP+

Why is BERT Effective for Ranking?





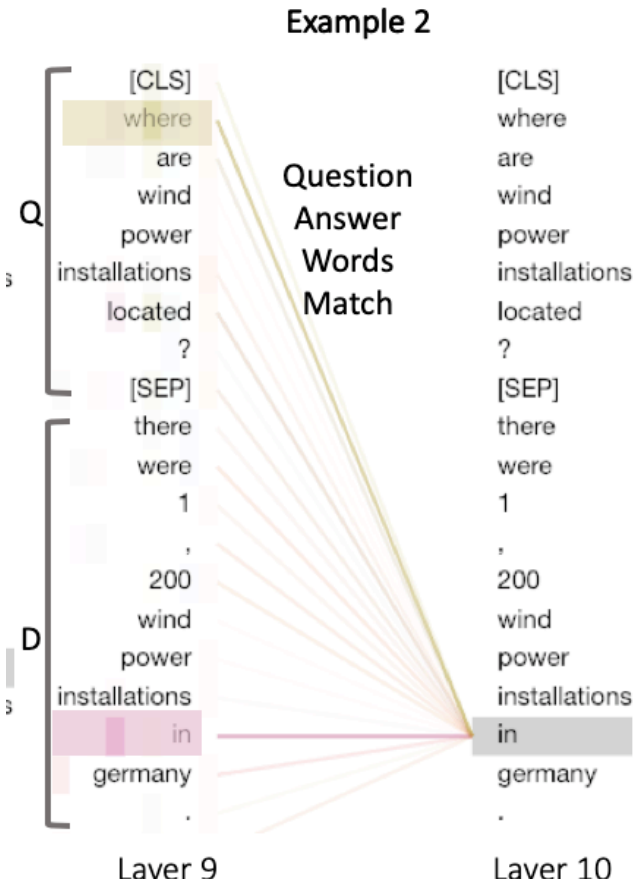
Match Words, Bigrams, Phrases (similar to KNRM/Conv-KNRM)

Q: "where are … "
D: "… in Germany"
(new to IR)

Why is BERT Effective for Ranking?

Traditionally, stopwords and punctuation are difficult for IR

- Noisy for **matching**
- Do not carry much information on their own
- Appear everywhere



Q: "where are … "
D: "… in Germany"
(new to IR)

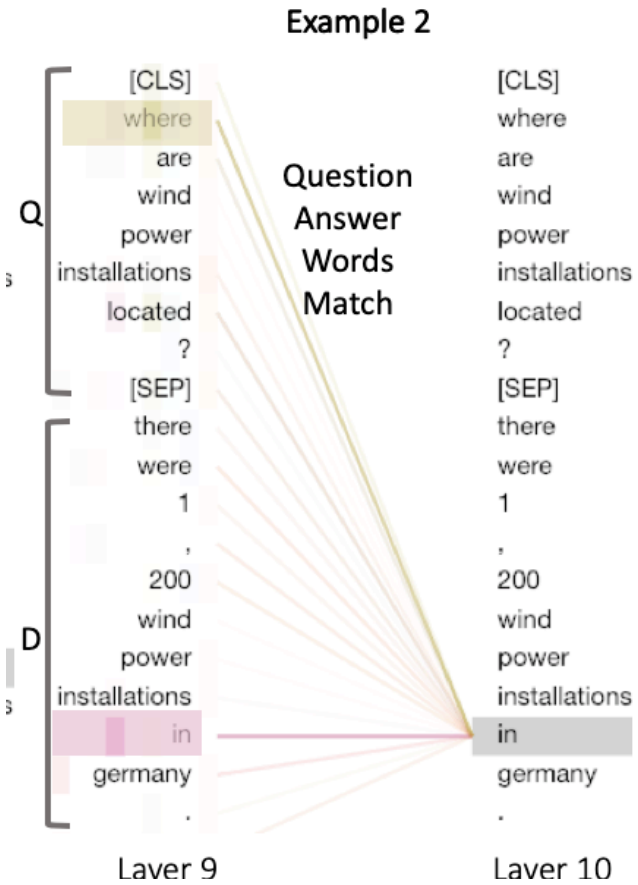
Why is BERT Effective for Ranking?

Traditionally, stopwords and punctuation are difficult for IR

- Noisy for **matching**
- Do not carry much information on their own
- Appear everywhere

But they improve DocBERT

- Important for **content understanding**
- Integral to the sentence structure



Q: "where are … "
D: "… in Germany"
(new to IR)

Summary

DocBERT Reranker

- A passage-based framework to use BERT for ranking

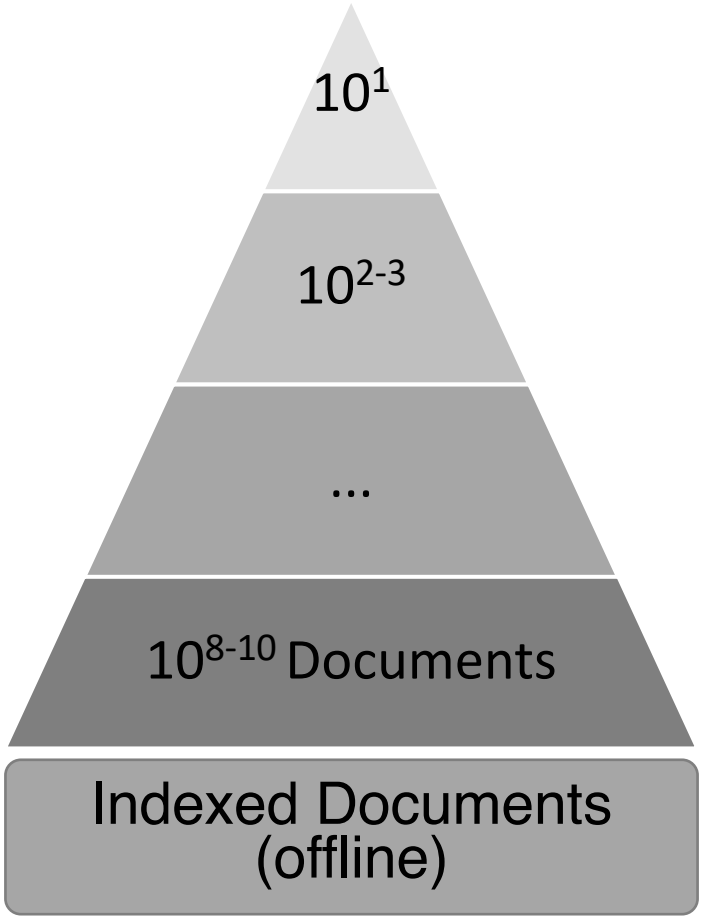
2 Types of Knowledge: General + Search-Specific

- General LMs perform reasonably well under low resource
- To get real improvements, we still need to learn search-specific knowledge from a large amount of search data

2 Types of Tasks: Matching + Understanding

- Match query tokens to document tokens
- Build context within a query and a document, addressing previously difficult IR problems
  - long queries, stopwords and punctuation

Challenges



PART I: MATCHING

Kernel-Based Ranking

Bring Soft Match into Ranking

BERT-Based Ranking

Bring General Language Understanding into Matching

INITIAL RETRIEVAL

Still counting terms and using heuristics (BM25, 1994)

Challenges

Apply DocBERT to Initial Retrieval?

Challenges

Apply DocBERT to Initial Retrieval?

500M docs X 1 query X 3.5ms* = 20.3 days

For One Query!

*: [Nogueira and Lin 2019]

Challenges

2 Types of Knowledge: General + Search-Specific

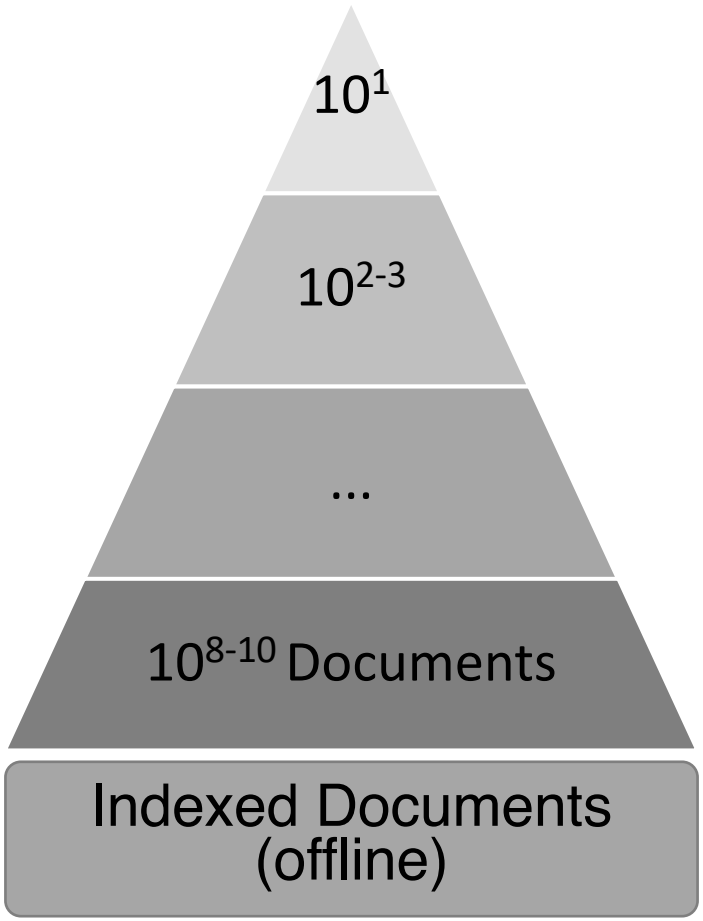
- General-purpose LMs performs reasonably well with limited training data
- To get real … d to learn search-specific kn … nt of search data

Can we move understanding to offline?

2 Types of Tasks: Matching + Understanding

- DocBERT matches query tokens to document tokens
- DocBERT also builds context within a query and a document, addressing previously difficult IR problems
  - long queries, stopwords and punctuation.

This Talk



PART I: MATCHING

Kernel-Based Ranking

Bring Soft Match into Ranking

BERT-Based Ranking

Bring General Language Understanding into Matching

PART II: REPRESENTATION

Context-Aware Term Weighting

Bring Deeper Language Understanding into Initial Retrieval

CONTEXT-AWARE TERM WEIGHTING

Bring Deeper Language Understanding into Initial Retrieval

[**D**C arXiv'19]
[**D**C SIGIR'20[s]]
[**D**C WWW'20]

[*]: Equal Contribution, [s]: Short Paper

Term Frequency *(tf)* != Importance

A Document

"Unlike cats, dogs are usually great exercise pals. Many breeds enjoy running and hiking, and will happily trek along on any trip. Exercise time varies…"

`{dog:1, cat:1, exercise:2, run:1, …}`

Frequency Weighting

Term Frequency *(tf)* != Importance

A Document

"Unlike cats, dogs are usually great exercise pals. Many breeds enjoy running and hiking, and will happily trek along on any trip. Exercise time varies…"

`{dog:1, cat:1, exercise:2, run:1, …}`

Frequency Weighting

How to let machines know the **essential meaning** of a document?

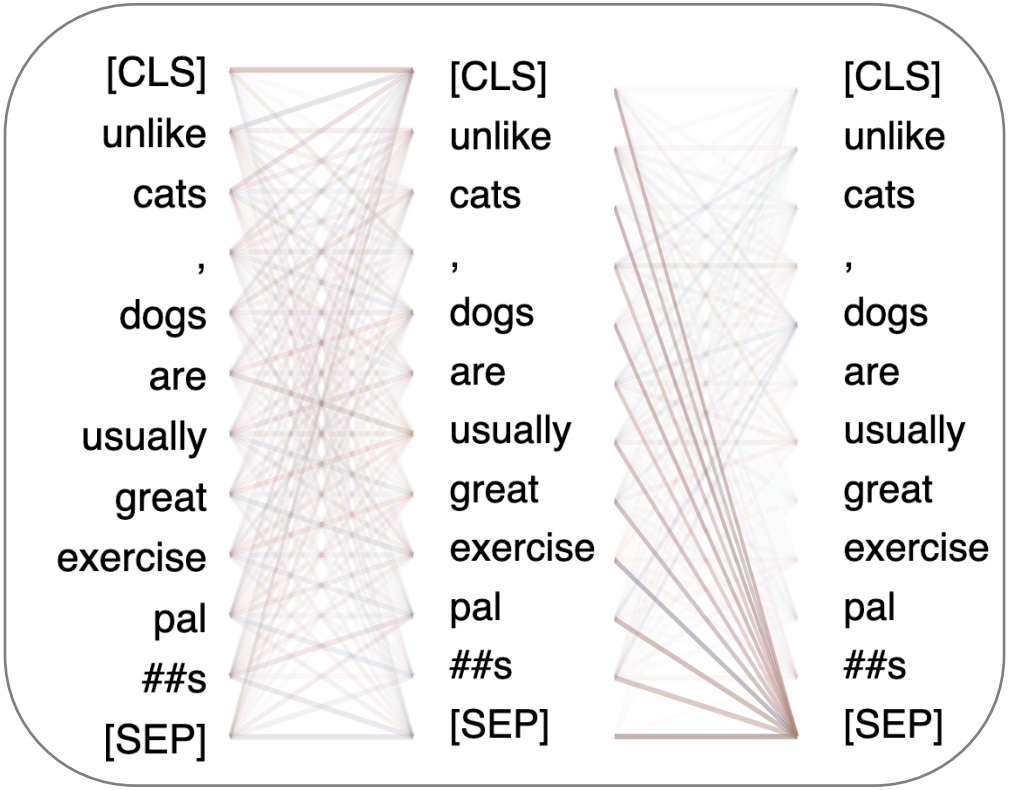
And **fast** enough for ranking millions of documents?

DeepCT:
Deep Context-Aware Term Weighting

- FAST: still store document as Bag-of-Words
- DEEP: understand word importance in its context

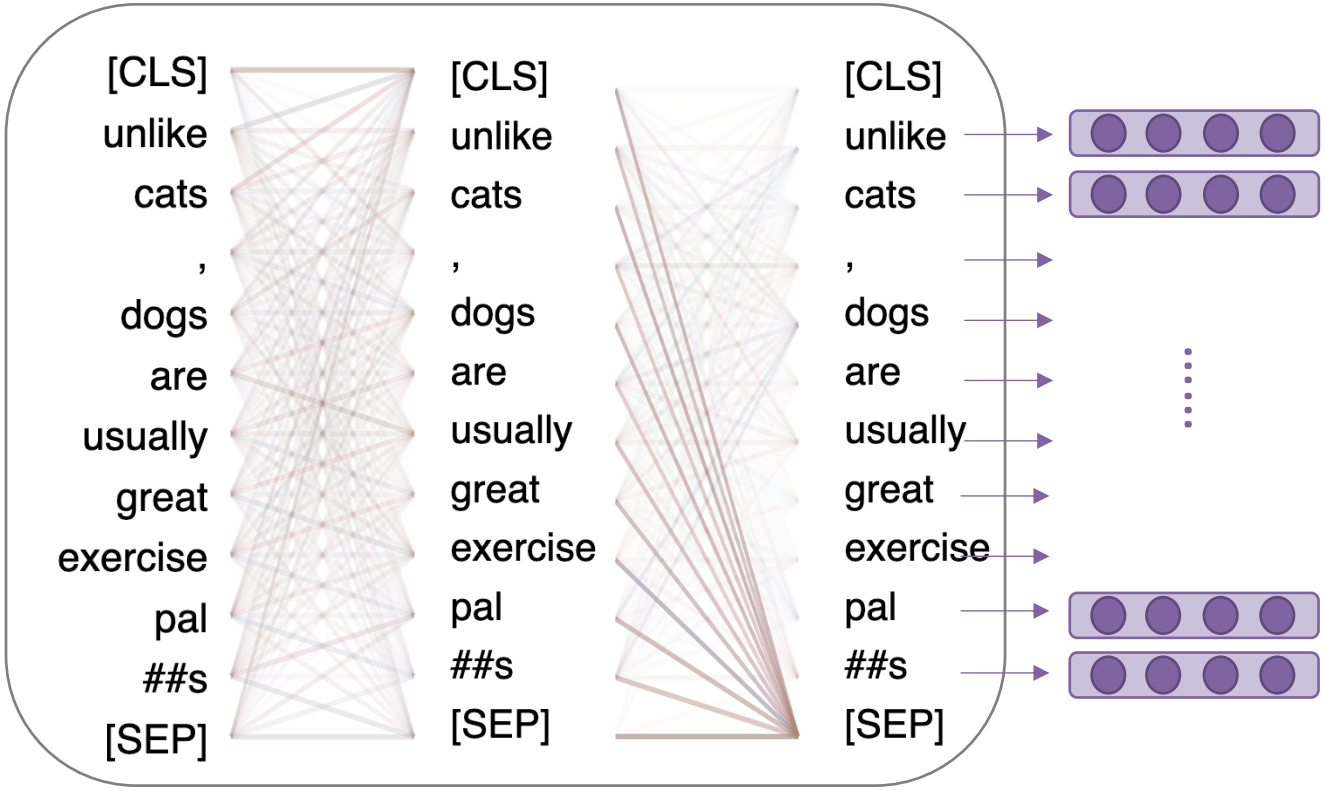
"Unlike cats, dogs are usually great exercise pals. Many breeds enjoy running and hiking, and will happily trek along on any trip. Exercise time varies…"

DeepCT:
Deep Context-Aware Term Weighting



Transformer:
Capture Context

DeepCT:
Deep Context-Aware Term Weighting

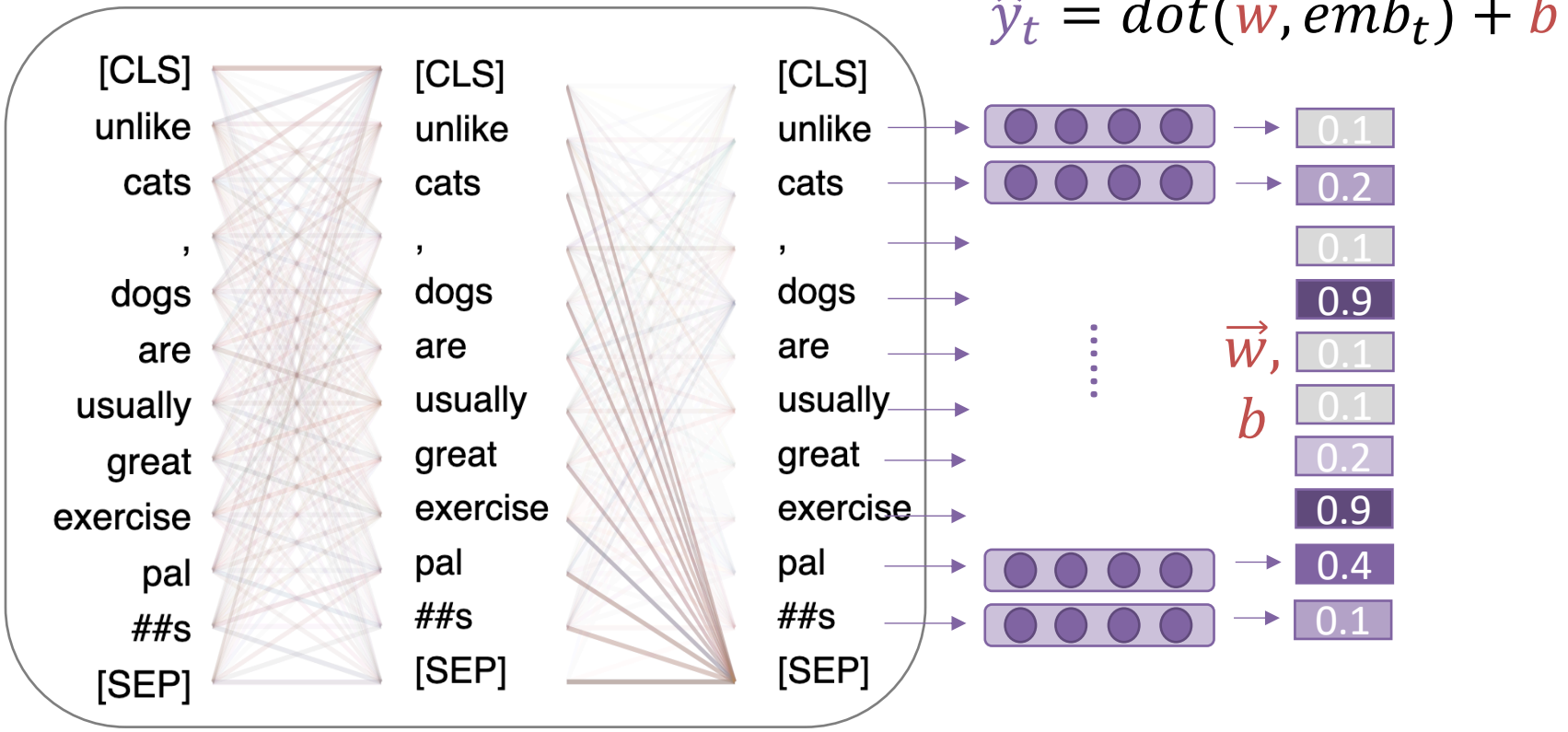


Transformer:
Capture Context

Contextualized
Term Embeddings

DeepCT:
Deep Context-Aware Term Weighting

$$\hat{y}_t = dot(\vec{w}, emb_t) + b$$



Transformer: Capture Context

Contextualized Term Embeddings

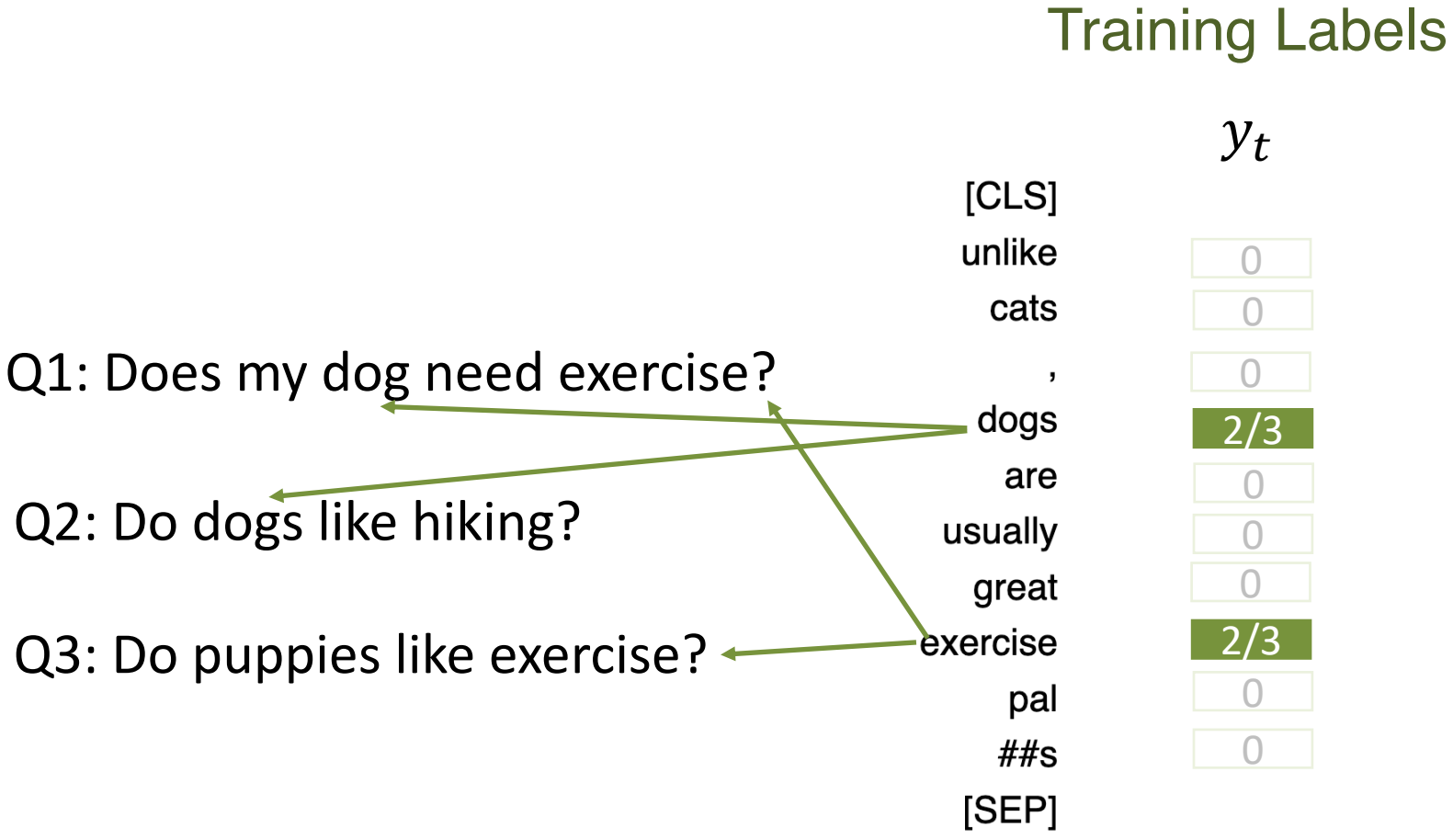
Context-Aware Term Weights

How to Train DeepCT

Training Labels

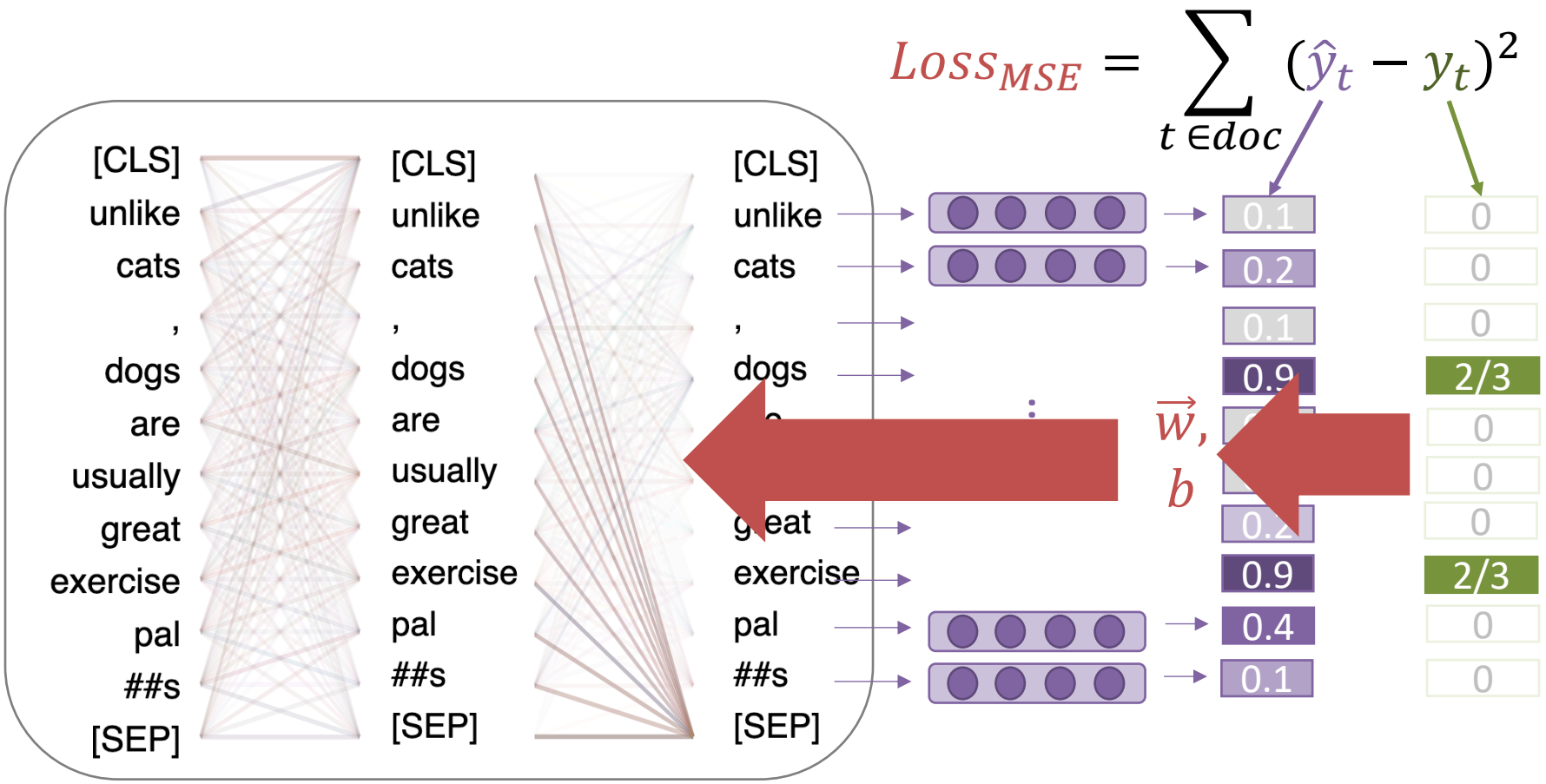
	y_t
[CLS]	
unlike	0
cats	0
,	0
dogs	2/3
are	0
usually	0
great	0
exercise	2/3
pal	0
##s	0
[SEP]	

How to Train DeepCT

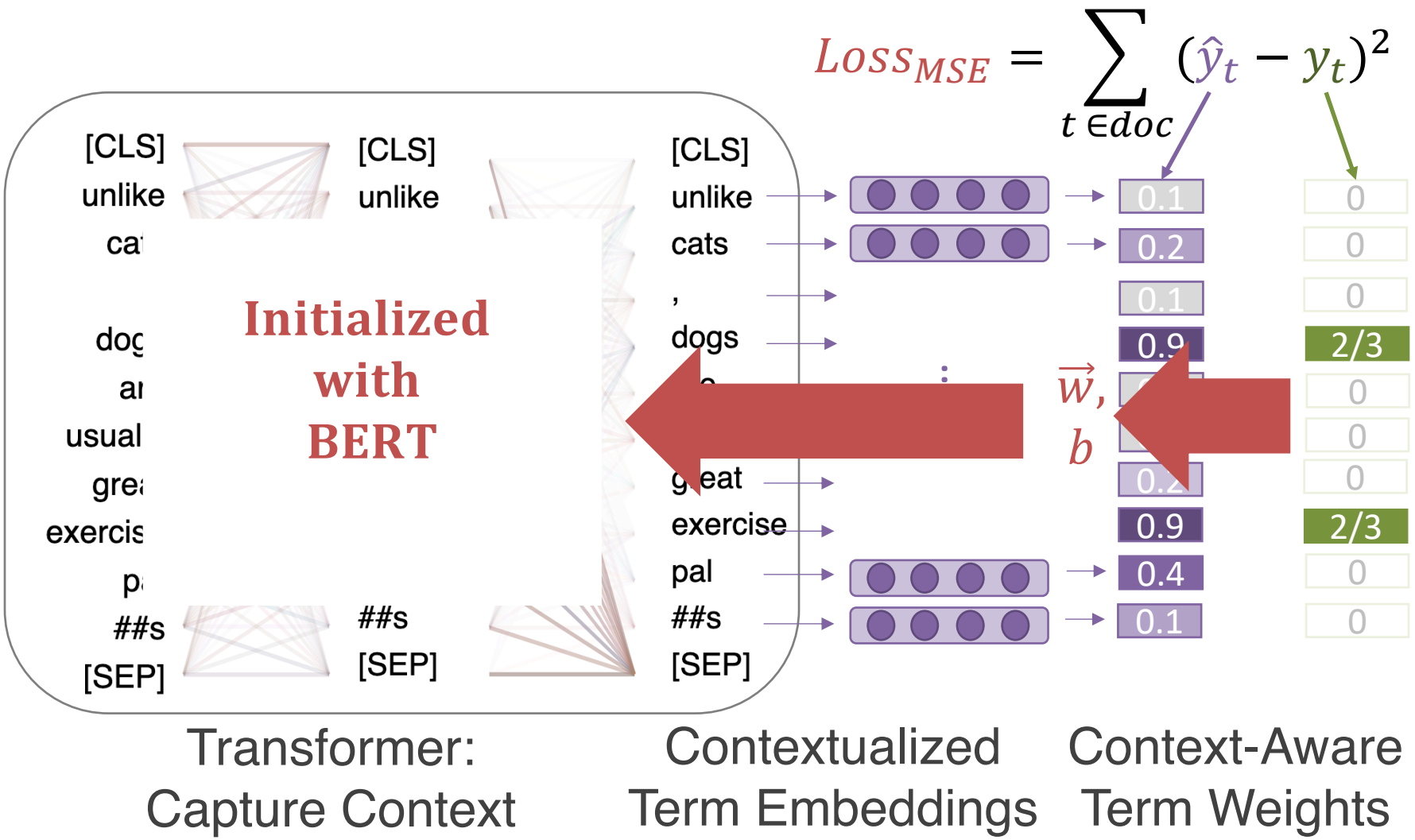


Train with $y_t = \frac{|Q_{d,t}|}{|Q_d|}$ (supervised)

How to Train DeepCT



How to Train DeepCT



Move Understanding to Offline, Keep Online Matching Simple

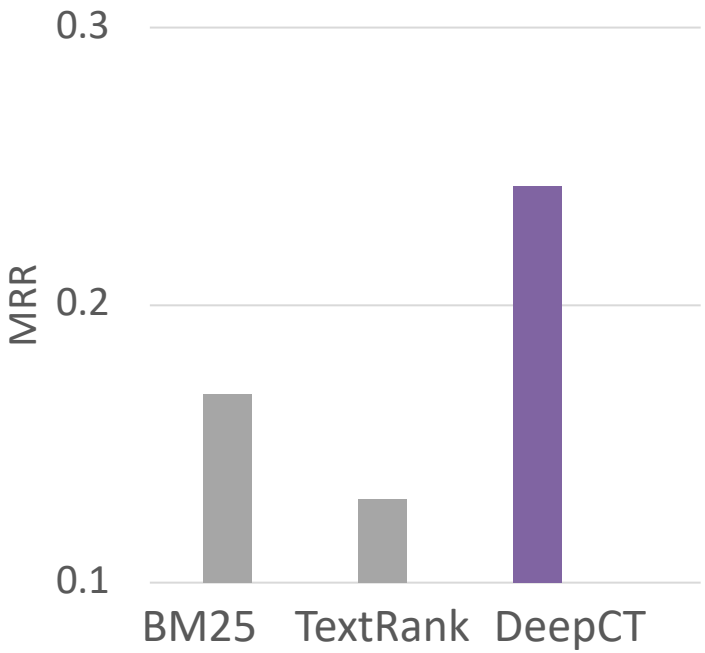
Offline:

- Apply DeepCT to all documents
- Transform scores to positive integer importance weights
  - Scores tend to fall in [0, 1]
  - $100x$, $\sqrt{100x}$, …
- In a typical inverted index, replace *tf* with DeepCT weights

Online:

- Rank with BM25 as usual

Effectiveness: Initial Ranking



DeepCT:

- +33% over BM25
- As fast as BM25

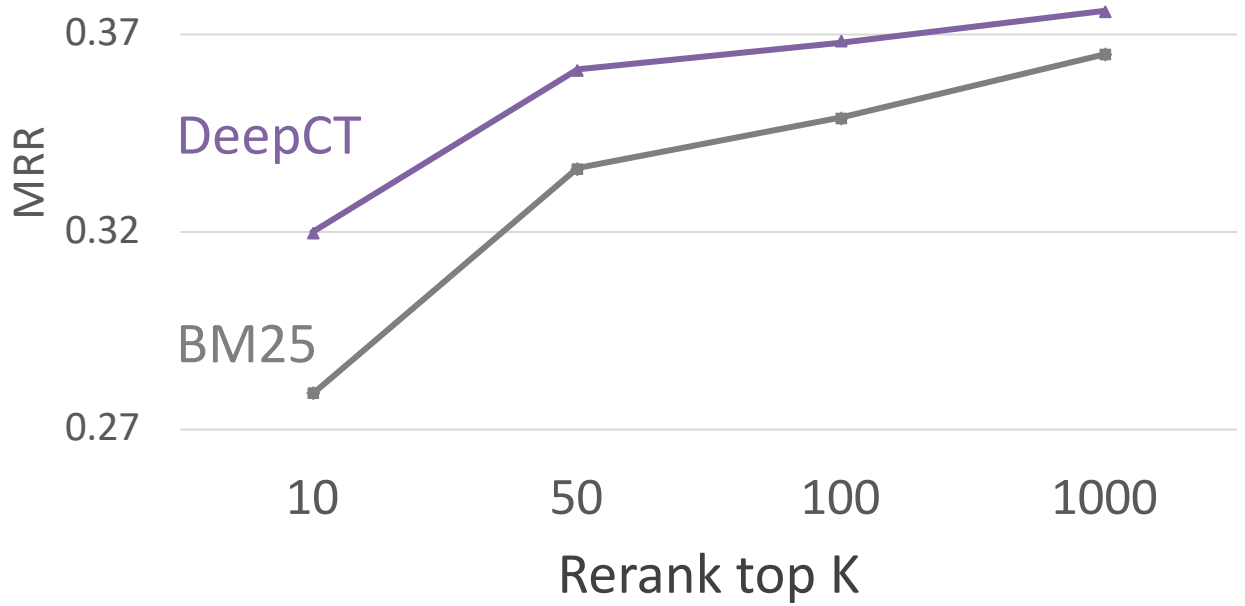
Dataset: MS MARCO Passages (8 million documents)

Training: 500K query-doc pairs

Testing: dev set (6980 queries)

Effectiveness: Used with Reranking

- Retrieval top K using BM25 or DeepCT
- Rerank top K using a SOTA BERT Reranker



Dataset: MS MARCO Passages
Training: 500K query-doc pairs
Testing: dev set (6980 queries)

*: [Nogueira and Lin 2019]

Effectiveness: Used with Reranking

	Rank	Model	Ranking Style	Submission Date	MRR@10 On Eval	MRR@10 On Dev
	1	**DR-BERT** X.W. S of Meituan-Dianping NLP-KG Group	Full Ranking	May 20th, 2020	0.419	0.420
	2	**expando-mono-duo-T5** Ronak Pradeep, Rodrigo Nogueira, Zhiying Jiang, and Jimmy Lin - University of Waterloo	Full Ranking	May 19th, 2020	0.408	0.420
DeepCT	3	**DeepCT + TF-Ranking Ensemble of BERT, ROBERTA and ELECTRA** (1) Shuguang Han, (2) Zhuyun Dai, (1) Xuanhui Wang, (1) Michael Bendersky and (1) Marc Najork - 1) Google Research, (2) Carnegie Mellon - Paper and Code	Full Ranking	June 2nd, 2020	0.407	0.421
	4	**UED** Anonymous	Full Ranking	May 5th, 2020	0.405	0.414
	5	**TABLE Model** X.W. S of Meituan-Dianping NLP-KG Group	Full Ranking	May 11th, 2020	0.401	0.412
DeepCT	6	**TABLE Model** X.W. S of Meituan-Dianping NLP-KG Group	Full Ranking	January 21th, 2020	0.400	0.401
	7	**TABLE Model** X.W. S of Meituan-Dianping NLP-KG Group	Full Ranking	May 8th, 2020	0.400	0.401
DeepCT	8	**DeepCT Retrieval + TF-Ranking BERT Ensemble** 1) Shuguang Han, (2) Zhuyun Dai, (1) Xuanhui Wang, (1) Michael Bendersky and (1) Marc Najork - (1) Google Research, (2) Carnegie Mellon University - Paper [Han, et al. '20]Code	Full Ranking	April 10th, 2020	0.395	0.405
DeepCT	9	**DeepCT + Bart** Binsheng Liu - RMIT University	Full Ranking	May 6th, 2020	0.394	0.408
	10	**Enriched BERT base + AOA index + CAS** Ming Yan of Alibaba Damo NLP	Full Ranking	August 20th, 2019	0.393	0.408

4 out of 10 best MS MARCO runs used DeepCT for initial retrieval. Date: 06/16/2020

HDCT: Extend DeepCT to Long Documents

Yellowstone National Park

From Wikipedia, the free encyclopedia

Coordinates: 44°36′N 110°30′W

"Yellowstone" redirects here. For other uses, see Yellowstone (disambiguation).

Yellowstone National Park is an American national park located mostly in Wyoming, with small sections in Montana and Idaho. It was established by the U.S. Congress and signed into law by President Ulysses S. Grant on March 1, 1872.[5][6] Yellowstone was the first national park in the U.S. and is also widely held to be the first national park in the world.[7] The park is known for its wildlife and its many geothermal features, especially Old Faithful geyser, one of its most popular features.[8] It has many types of ecosystem, but the subalpine forest is the most abundant. It is part of the South Central Rockies forests ecoregion.

Native Americans have lived in the Yellowstone region for at least 11,000 years.[9] Aside from visits by mountain men during the early-to-mid-19th century, organized exploration did not begin until the late 1860s. Management and control of the park originally fell under the jurisdiction of the Secretary of the Interior, the first being Columbus Delano. However, the U.S. Army was subsequently commissioned to oversee management of Yellowstone for a 30-year period between 1886 and 1916.[10] In 1917, administration of the park was transferred to the National Park Service, which had been created the previous year. Hundreds of structures have been built and are protected for their architectural and historical significance, and researchers have examined more than a thousand archaeological sites.

Yellowstone National Park spans an area of 3,468.4 square miles (8,983 km^2),[2] comprising lakes, canyons, rivers and mountain ranges.[8] Yellowstone Lake is one of the largest high-elevation lakes in North America and is centered over the Yellowstone Caldera, the largest supervolcano on the continent. The caldera is considered a dormant volcano. It has erupted with tremendous force several times in the last two million years.[11] Half of the world's geysers[12][13] and hydrothermal features[14] are in Yellowstone, fueled by this ongoing volcanism. Lava flows and rocks from volcanic eruptions cover most of the land area of Yellowstone. The park is the centerpiece of the Greater Yellowstone Ecosystem, the largest remaining nearly-intact ecosystem in the Earth's northern temperate zone.[15] In 1978, Yellowstone was named a UNESCO World Heritage Site.

Hundreds of species of mammals, birds, fish, and reptiles have been documented, including several that are either endangered or threatened.[8] The vast forests and grasslands also include unique species of plants. Yellowstone Park is the largest and most famous megafauna location in the contiguous United States. Grizzly bears, wolves, and free-ranging herds of bison and elk live in this park. The Yellowstone Park bison herd is the oldest and largest public bison herd in the United States. Forest fires occur in the park each year; in the large forest fires of 1988, nearly one third of the park was burnt. Yellowstone has numerous recreational opportunities, including hiking, camping, boating, fishing and sightseeing. Paved roads provide close access to the major geothermal areas as well as some of the lakes and waterfalls. During the winter, visitors often access the park by way of guided tours that use either snow coaches or snowmobiles.

Yellowstone National Park
IUCN category II (national park)[1]

Coordinates	44°36′N 110°30′W
Area	2,219,791 acres (8,983.18 km^2)[2]
Established	March 1, 1872
Visitors	4,115,000 (in 2018)[3]
Governing body	U.S. National Park Service
Website	Official website
UNESCO World Heritage Site	
Type	Natural
Criteria	vii, viii, ix, x
Designated	1978 (2nd session)
Reference no.	28[4]
Region	The Americas
Endangered	1995–2003

History [edit]

Detailed pictorial map from 1904

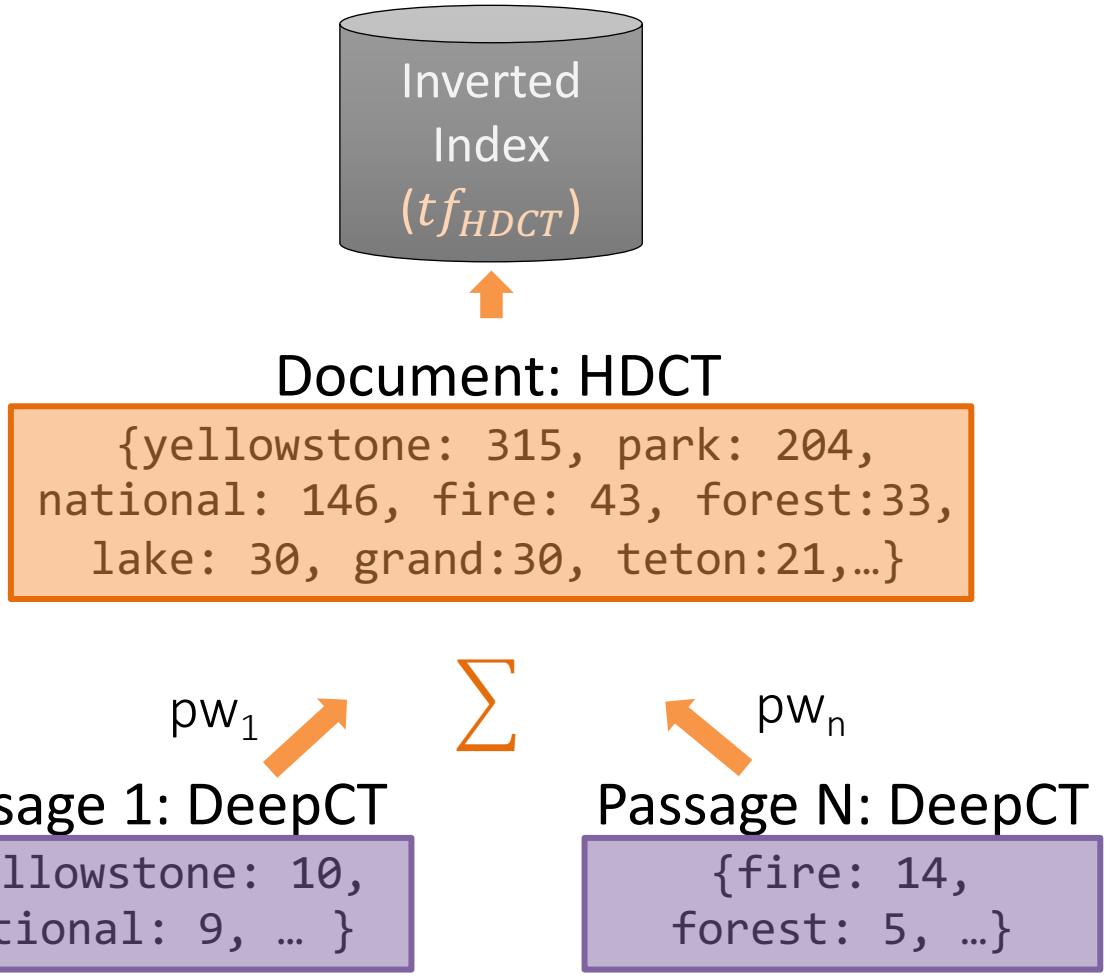
The park contains the headwaters of the Yellowstone River, from which it takes its historical name. Near the end of the 18th century, French trappers named the river *Roche Jaune*, which is probably a translation of the Hidatsa name *Mi tsi a-da-zi* ("Yellow Rock River").[16] Later, American trappers rendered the French name in English as "Yellow Stone". Although it is commonly believed that the river was named for the yellow rocks seen in the Grand Canyon of the Yellowstone, the Native American name source is unclear.[17]

The human history of the park began at least 11,000 years ago when Native Americans began to hunt and fish in the region. During the construction of the post office in Gardiner, Montana, in the 1950s, an obsidian projectile point of Clovis origin was found that dated from approximately 11,000 years ago.[18] These Paleo-Indians, of the Clovis culture, used the significant amounts of obsidian found in the park to make cutting tools and weapons. Arrowheads made of Yellowstone obsidian have been found as far away as the Mississippi Valley, indicating that a regular obsidian trade existed between local tribes and tribes farther east.[19] By the time white explorers first entered the region during the Lewis and Clark Expedition in 1805, they encountered the Nez Perce, Crow, and Shoshone tribes. While passing through present day Montana, the expedition members heard of the Yellowstone region to the south, but they did not investigate it.[19]

In 1806, John Colter, a member of the Lewis and Clark Expedition, left to join a group of fur trappers. After splitting up with the other trappers in 1807, Colter passed through a portion of what later became the park, during the winter of 1807–1808. He observed at least one geothermal area in the northeastern section of the park, near Tower Fall.[20] After surviving wounds he suffered in a battle with members of the Crow and Blackfoot tribes in 1809, Colter described a place of "fire and brimstone" that most people dismissed as delirium; the supposedly imaginary place was nicknamed "Colter's Hell". Over the next 40 years, numerous reports from mountain men and trappers told of boiling mud, steaming rivers, and petrified trees, yet most of these reports were believed at the time to be myth.[21]

After an 1856 exploration, mountain man Jim Bridger (also believed to be the first or second European American to have seen the Great Salt Lake) reported observing boiling springs, spouting water, and a mountain of glass and yellow rock. These reports were largely ignored because Bridger was a known "spinner of yarns". In 1859, a U.S. Army Surveyor named Captain William F. Raynolds embarked on a two-year survey of the northern Rockies. After wintering in Wyoming, in May 1860, Raynolds and his party—which included

HDCT: Context-Aware Hierarchical Document Term Weighting

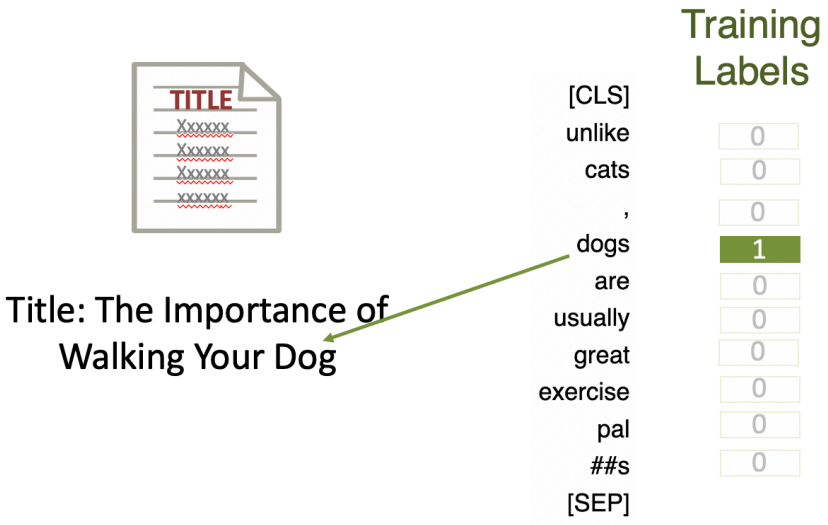


Unsupervised Training

Unsupervised w/ Title:

$$y_{t,p} = \begin{matrix} 1 & t \in d_{title} \\ 0 & otherwise \end{matrix}$$

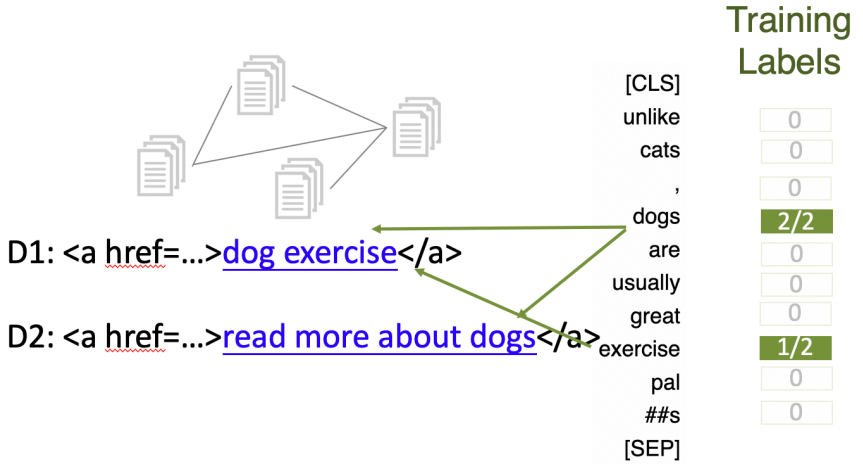
- Internal structure in the document



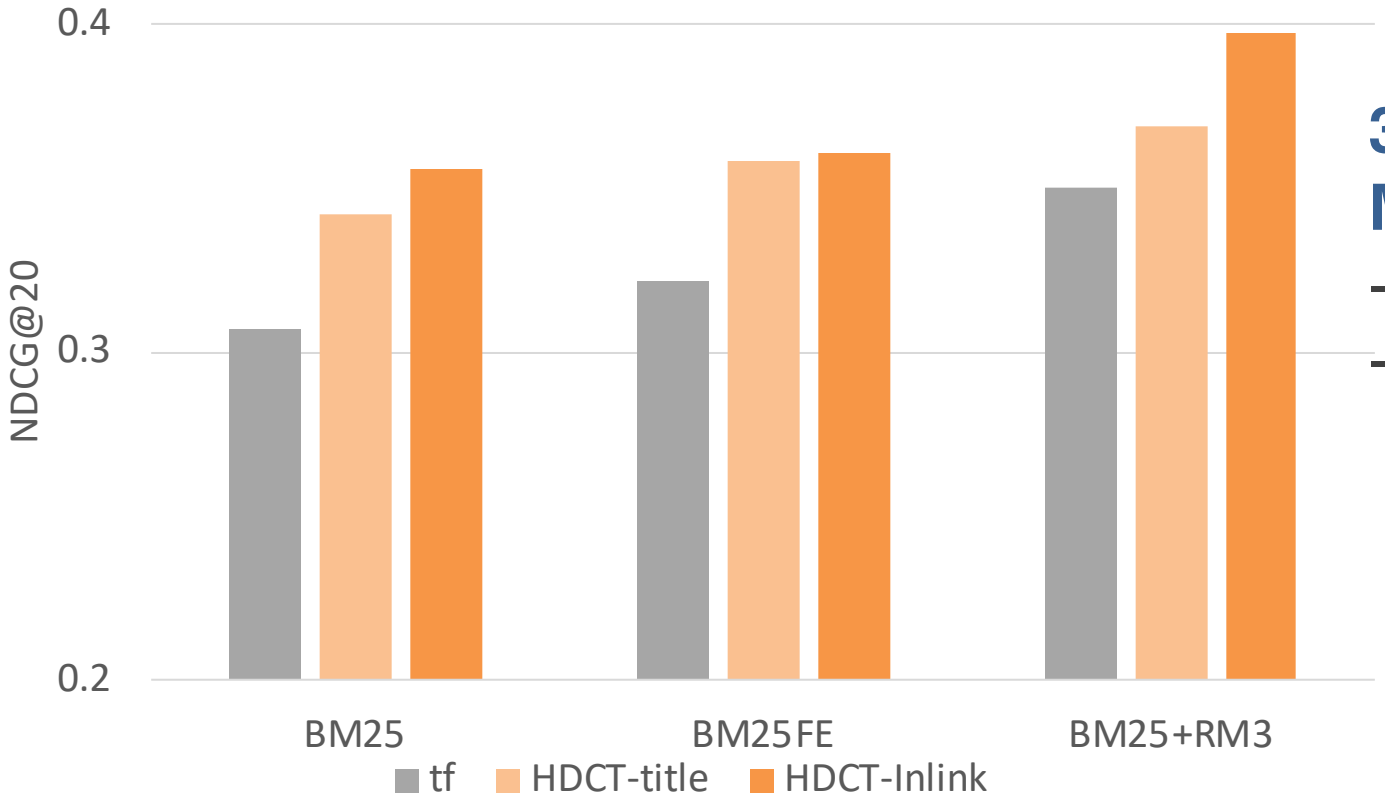
Unsupervised w/ Inlinks:

$$y_{t,p} = \frac{|Q_{d,t}|}{|Q_d|}$$

- Graph structure across documents



HDCT Effectiveness



3 Retrieval Models:

- HDCT > *tf*
- Inlink > Title

Dataset: ClueWeb09-C (5M documents)

Testing: 200 queries from TREC (title)

Summary

A new approach to estimating term importance

- From “frequencies” to “meanings”
- Substantially improve *tf*-based retrieval baselines
- Extend to longer documents & low-resource domains

It is time to change the initial retrieval models

- *tf*-based retrieval: used for 50-60 years
- Now, it can be replaced
- Better text understanding & representation
  => better initial retrieval
  => better end-to-end effectiveness/efficiency

CONCLUSION

My dissertation research develops a suite of neural network based approaches to addressing two critical bottlenecks in information retrieval

Query-Document Matching:

Exact Lexical Match to Soft Match

Query/document Representation:

Frequencies to Meanings

	Retrieval	Reranking	MRR@10
Dissertation PART II	BM25	Learning-To-Rank* (Official Baseline)	0.195
		K-NRM* (Chap. 3)	0.218
		Conv-KNRM* (Chap. 4)	0.247
		Conv-KNRM ensemble* (Chap. 3)	0.290
		DocBERT Reranker (Chap. 6)	0.364
Dissertation PART III	BM25* (Official Baseline)	None	0.167
	DeepCT (Chap. 7)		0.243
	DeepCT+BM25FE (Chap. 8)		0.250
Combined	DeepCT (Chap. 7)	Conv-KNRM (Chap. 4)	0.278
	DeepCT (Chap. 7)	BERT Reranker (Nogueira and Cho, 2019)	0.376
	DeepCT+BM25FE (Chap. 8)	DocBERT Reranker (Chap. 6)	0.394
	DeepCT (Chap. 7)	TF Ranking (Najork et al., 2020)	0.405

Provides a new view of how languages should be modeled in IR

Identified the Importance of Search-Specific Knowledge

- Similarity != Relevance

General Knowledge from Corpus Analysis

Reasonable Performance

Search-Specific Knowledge from Search Logs

Real Improvements

Provides a new view of how languages should be modeled in IR

Identified the Importance of Search-Specific Knowledge

- Similarity != Relevance

General Knowledge from Corpus Analysis

Reasonable Performance

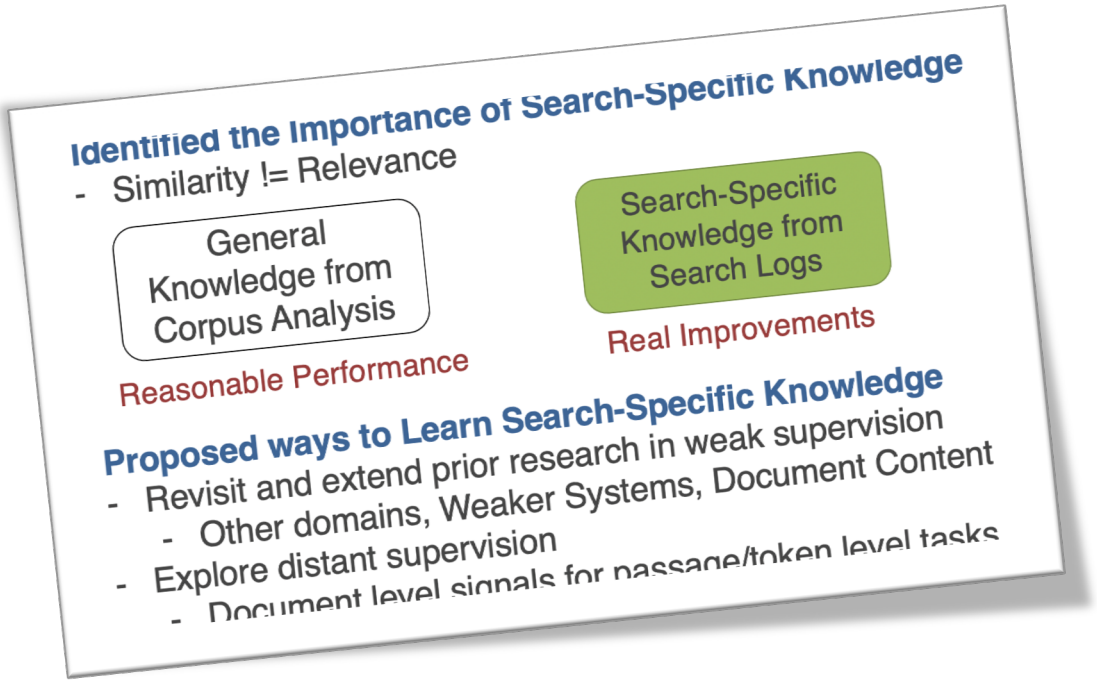
Search-Specific Knowledge from Search Logs

Real Improvements

Proposed ways to Learn Search-Specific Knowledge

- Revisit and extend prior research in weak supervision
  - Other domains, Weaker Systems, Document Content
- Explore distant supervision
  - Document level signals for passage/token level tasks

Provides a new view of how languages should be modeled in IR



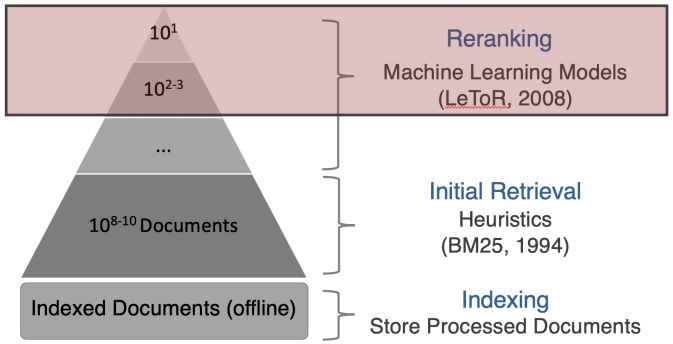
Future Direction:
Zero-shot/Few-shot IR (e.g., enterprise search, COVID-19)

Broadens the scope of Neural IR

Neural Matching

Neural Matching Models

DSSM (Huang et al, 2013), CDSSM (Shen et al, 2014), ARC-II (Hu et al, 2014), DRMM (Guo et al, 2016),

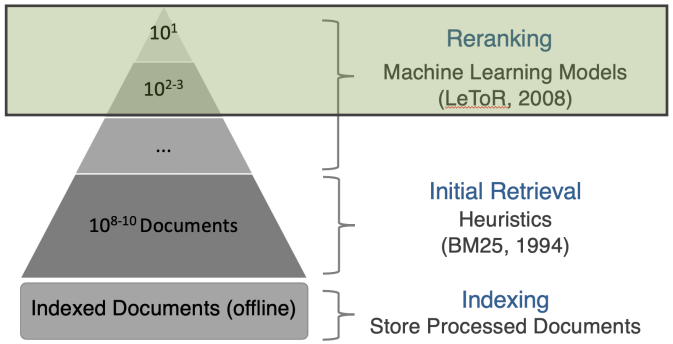


Broadens the scope of Neural IR

Neural Matching

Neural Matching Models

DSSM (Huang et al, 2013), CDSSM (Shen et al, 2014), ARC-II (Hu et al, 2014), DRMM (Guo et al, 2016), K-NRM, Conv-KNRM, Duet (Mitra et al, 2017), NRM-F(Zamani et al, 2018), …DocBERT, …

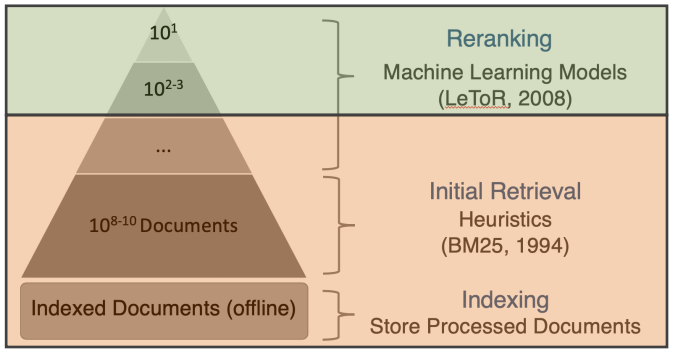


Broadens the scope of Neural IR

Neural Matching => Matching + Understanding

Neural Matching Models

DSSM (Huang et al, 2013), CDSSM (Shen et al, 2014), ARC-II (Hu et al, 2014), DRMM (Guo et al, 2016), K-NRM, Conv-KNRM, Duet (Mitra et al, 2017), NRM-F(Zamani et al, 2018), …DocBERT, …



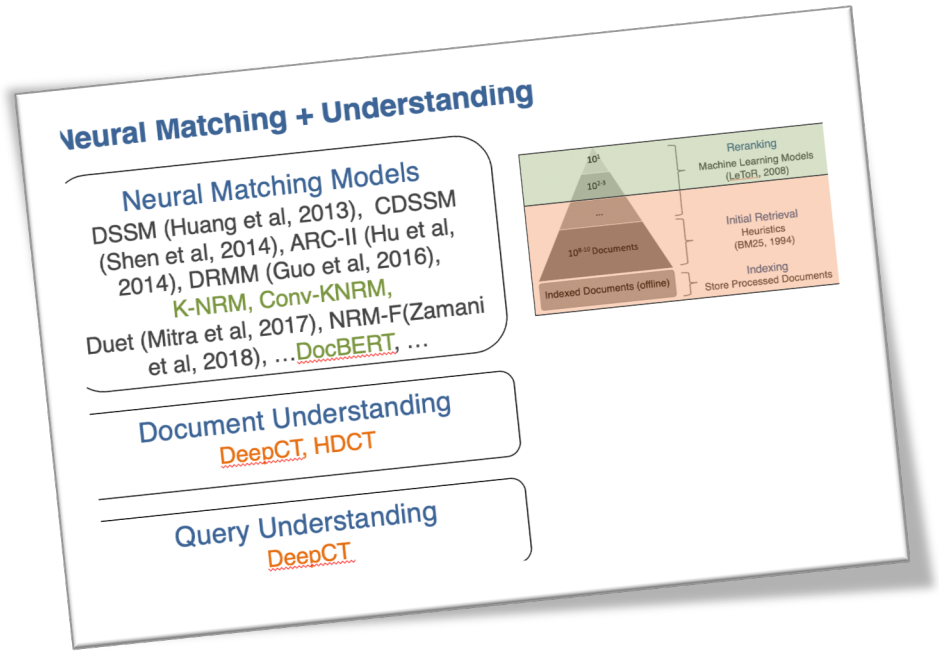
Document Understanding

DeepCT, HDCT

Query Understanding

DeepCT

Broadens the scope of Neural IR

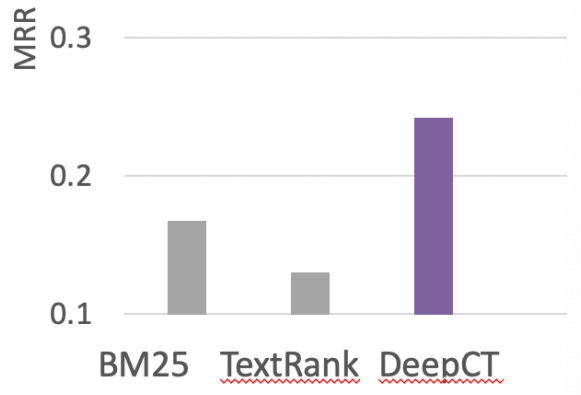


Future Direction:
Retrieval => Result Presentation
(e.g., conversational results summarization)

Changes how people will build retrieval pipelines

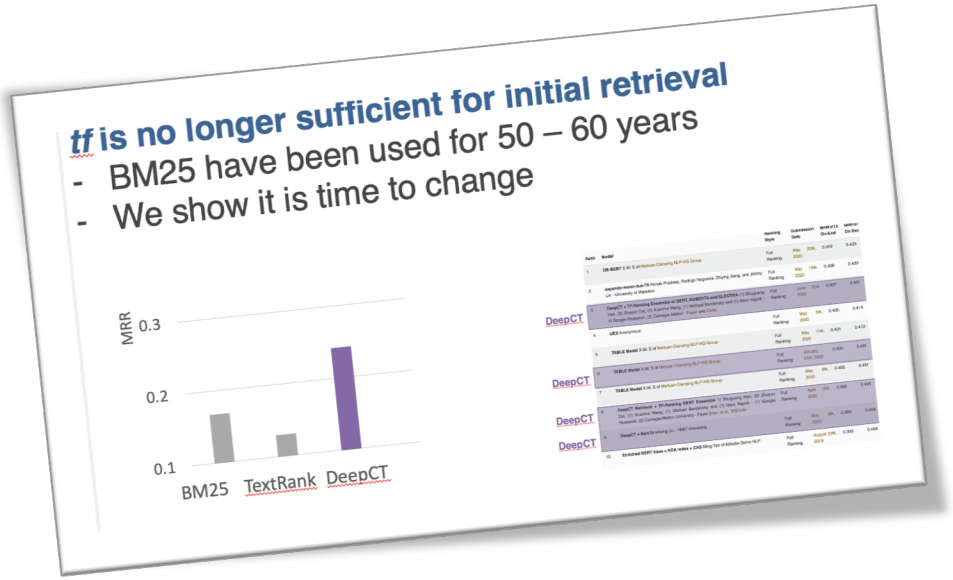
_tf_ is no longer sufficient for initial retrieval

- *tf* have been used for 50 – 60 years
- We show that it is time to change



	Rank	Model	Ranking Style	Submission Date	MRR@10 On Eval	MRR@10 On Dev
	1	**DR-BERT** X.W. S of Meituan-Dianping NLP-KG Group	Full Ranking	May 20th, 2020	0.419	0.420
	2	**expando-mono-duo-T5** Ronak Pradeep, Rodrigo Nogueira, Zhiying Jiang, and Jimmy Lin - University of Waterloo	Full Ranking	May 19th, 2020	0.408	0.420
DeepCT	3	**DeepCT + TF-Ranking Ensemble of BERT, ROBERTA and ELECTRA** (1) Shuguang Han, (2) Zhuyun Dai, (1) Xuanhui Wang, (1) Michael Bendersky and (1) Marc Najork - 1) Google Research, (2) Carnegie Mellon - Paper and Code	Full Ranking	June 2nd, 2020	0.407	0.421
	4	**UED** Anonymous	Full Ranking	May 5th, 2020	0.405	0.414
	5	**TABLE Model** X.W. S of Meituan-Dianping NLP-KG Group	Full Ranking	May 11th, 2020	0.401	0.412
DeepCT	6	**TABLE Model** X.W. S of Meituan-Dianping NLP-KG Group	Full Ranking	January 21th, 2020	0.400	0.401
	7	**TABLE Model** X.W. S of Meituan-Dianping NLP-KG Group	Full Ranking	May 8th, 2020	0.400	0.401
DeepCT	8	**DeepCT Retrieval + TF-Ranking BERT Ensemble** 1) Shuguang Han, (2) Zhuyun Dai, (1) Xuanhui Wang, (1) Michael Bendersky and (1) Marc Najork - (1) Google Research, (2) Carnegie Mellon University - Paper [Han, et al. '20]Code	Full Ranking	April 10th, 2020	0.395	0.405
DeepCT	9	**DeepCT + Bart** Binsheng Liu - RMIT University	Full Ranking	May 6th, 2020	0.394	0.408
	10	**Enriched BERT base + AOA index + CAS** Ming Yan of Alibaba Damo NLP	Full Ranking	August 20th, 2019	0.393	0.408

Changes how people will build retrieval pipelines



Future Direction:
"Now we know it works, it is to make it work better."

THANK YOU!

retrieval
query
neural
information
search
features
term
weights
text
match
data
soft
HDCT
DeepCT
dataset
results
accuracy
networks
labels
two
terms
index
signals
deep
pairs
first
BERT
DISH-Conv
knowledge
kernel
initial
Neural
shows
systems
layer
new
set
MRR
ACM
SIGIR
Figure
regions
experiments

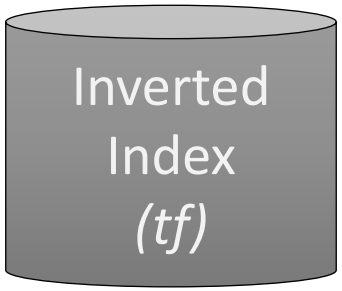
Offline Indexing & Initial Ranking

Offline:
Representation & Index

Online:
Look-Up & Rank

Document
Bag-of-Words

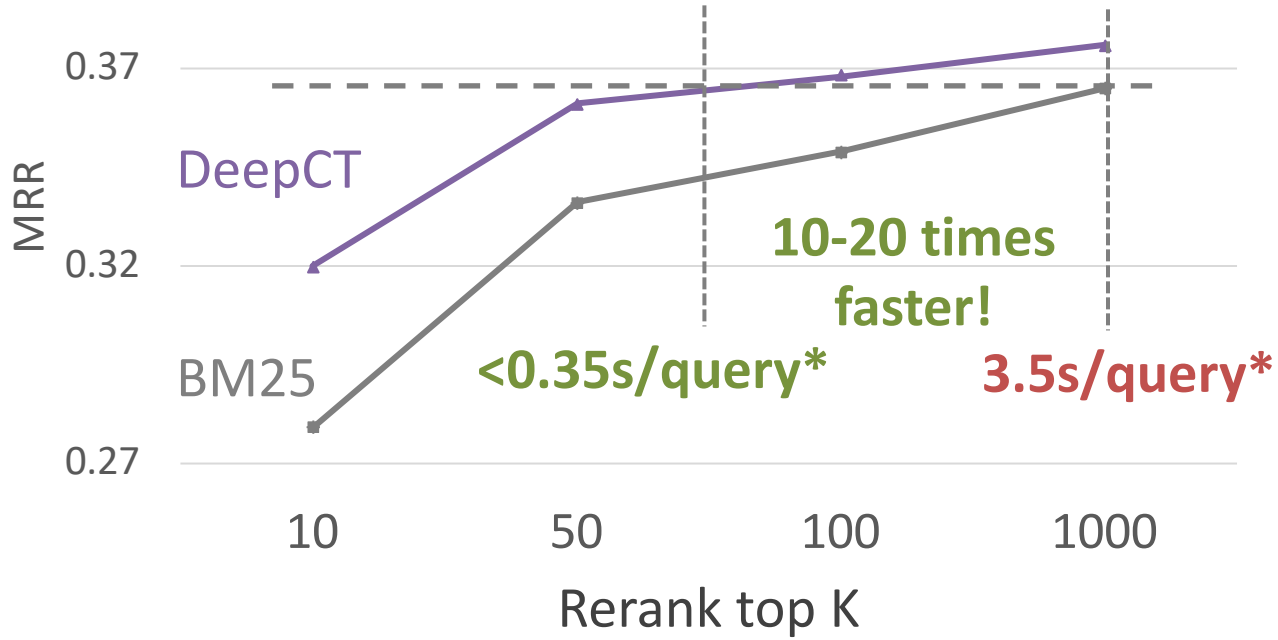
```
{cat:1,dog:1,
 exercise:2,
run:1,hike:1,
…}
```



$$\text{score}(q,d) = \sum_{t \text{ in } q} \text{H}(tf_{t,d}, idf_t)$$

Effectiveness: Used with Reranking

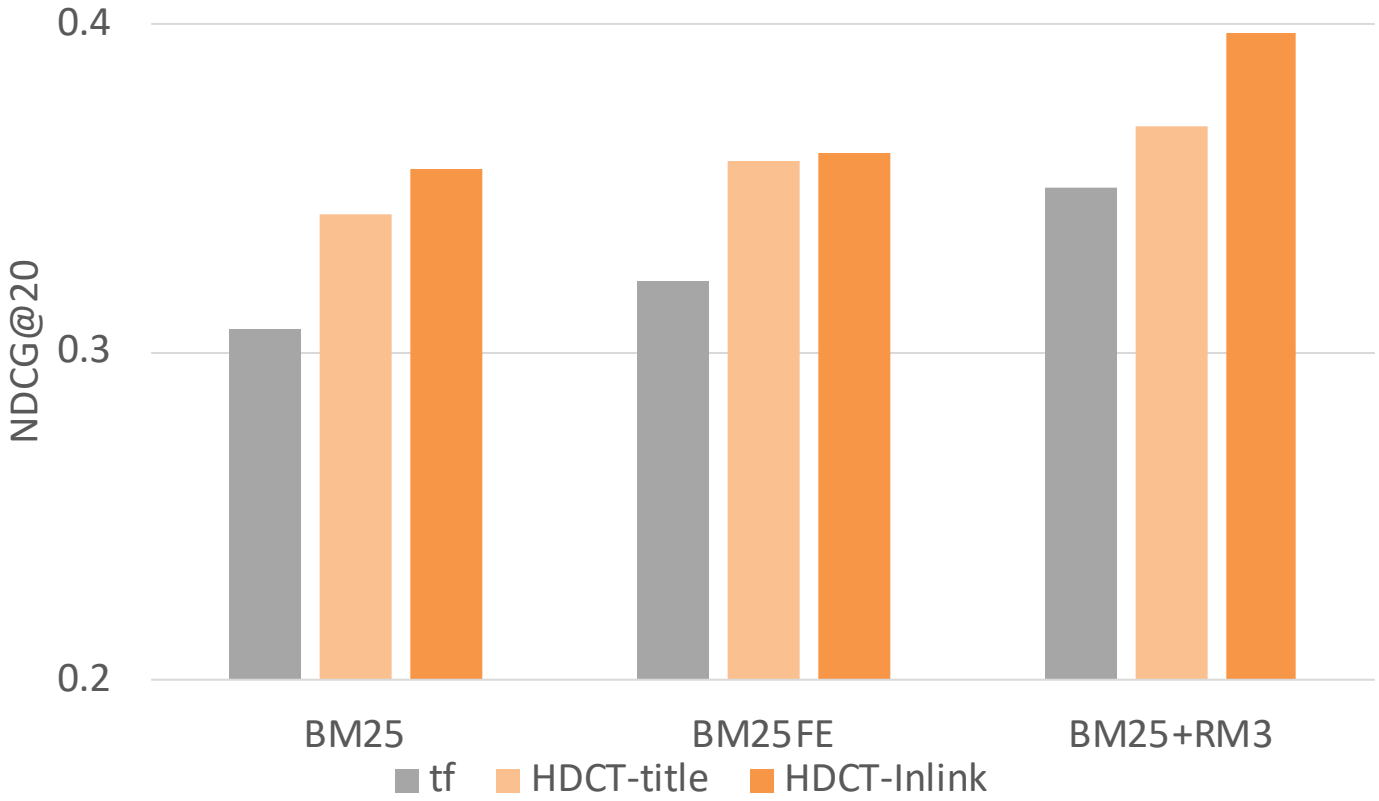
- Retrieval top K using BM25 or DeepCT
- Rerank top K using a SOTA BERT Reranker



Dataset: MS MARCO Passages
Training: 500K query-doc pairs
Testing: dev set (6980 queries)

*: [Nogueira and Lin 2019]

HDCT Effectiveness



RM3:

- A widely-used query expansion method
- HDCT also improves RM3

Dataset: ClueWeb09-C (5M documents)

Testing: 200 queries from TREC (title)